



“UNIVERSIDAD NACIONAL DE PIURA”

FACULTAD DE CIENCIAS

**ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA
Y TELECOMUNICACIONES**



**“SISTEMA DE MONITOREO DE LA CONTAMINACIÓN
AMBIENTAL, PRODUCIDA POR VEHÍCULOS DE TRANSPORTE
PÚBLICO”**

**PRESENTADA POR:
BACH. JUAN MANUEL LITANO VILCHEZ**

**TESIS PARA OBTAR EL TÍTULO PROFESIONAL DE INGENIERO
ELECTRÓNICO Y TELECOMUNICACIONES**

**PIURA – PERÚ
2015**

7483
LIT



**TESIS PRESENTADA COMO REQUISITO PARA OPTAR EL TÍTULO DE
INGENIERO ELECTRÓNICO Y TELECOMUNICACIONES**

ASESOR:



ING. FRANKLIN BARRA ZAPATA

TESISTA:



BACH. JUAN MANUEL LITANO VILCHEZ

JURADO EVALUADOR:

PRESIDENTE:



ING. EDWIN ARNALDO OCAS INFANTE

SECRETARIO:



ING. EDUARDO OMAR ÁVILA REGALADO

VOCAL:



ING. AYAX MANUEL SIFUENTES MONTES



DEDICATORIA

A mis padres, hermanas y sobrino por su apoyo incondicional durante mi formación profesional

AGRADECIMIENTO

Esta tesis no hubiera sido posible sin el impulso de todas aquellas personas que me han ayudado personalmente en el desarrollo del trabajo de esta índole, comenzando por el Ing. Franklin Barra Zapata a quien agradezco su apoyo en la elaboración del presente sistema de monitoreo.

Un sincero agradecimiento al jefe del área de trabajo, por darme permiso siempre que lo requiera.

Un agradecimiento muy especial por la paciencia y el ánimo recibidos de mi familia y amigos.

A todos ellos, muchas gracias.

INDICE	PÁG.
ÍNDICE DE TABLAS	X
ÍNDICE DE FIGURAS	XI
ANEXOS	XIII
RESUMEN	XIV
ABSTRACT	XV
INTRODUCCIÓN	16

CAPÍTULO I

1.0 DESCRIPCIÓN DE LA REALIDAD PROBLEMÁTICA	17
1.1 DELIMITACIONES Y DEFINICIÓN DEL PROBLEMA	17
1.1.1 DELIMITACIONES	17
1.1.2 DEFINICIÓN DEL PROBLEMA	17
1.2 FORMULACIÓN DEL PROBLEMA	18
1.2.1 PROBLEMA PRINCIPAL	18
1.3 OBJETIVOS DE LA INVESTIGACIÓN	18
1.3.1 OBJETIVO GENERAL	18
1.3.2 OBJETIVOS ESPECÍFICOS	18
1.4 HIPÓTESIS GENERAL	18
1.4.1 HIPÓTESIS GENERAL	18

CAPÍTULO II

2.0 MARCO TEÓRICO	19
2.1 ANTECEDENTES DE LA INVESTIGACIÓN	19
2.2 MARCO CONCEPTUAL	20
2.2.1 MICROCONTROLADORES	20
2.2.1.1 ¿QUÉ ES UN MICROCONTROLADOR?	20
2.2.1.2 ¿QUE PUEDO HACER CON EL MICROCONTROLADOR?	20
2.2.1.2.1 PRINCIPALES FABRICANTES	21
2.2.1.3 APARICIÓN Y DESARROLLO DE LOS MICROCONTROLADORES	22
2.2.1.3.1 BREVE ESBOZO HISTÓRICO	23
2.2.1.4 ARQUITECTURA DE UN MICROCONTROLADOR	23
2.2.1.4.1 ARQUITECTURA VON NEUMANN	23

2.2.1.4.2 ARQUITECTURA HARVARD	23
2.2.1.5 MICROCONTROLADOR A ELEGIR	24
2.2.1.5.1 ¿QUÉ MICROCONTROLADOR ELEGIR?	24
2.2.1.6 EL MICROCONTROLADOR 16F877A	25
2.2.1.6.1 CARACTERÍSTICAS PRINCIPALES	26
2.2.1.6.2 CARACTERÍSTICAS PERIFÉRICAS	26
2.2.1.6.3 CONFIGURACIÓN DE PINES	27
2.2.1.6.4 DESCRIPCIÓN DE LOS PINES DEL MICROCONTROLADOR	29
2.2.1.6.5 ARQUITECTURA INTERNA DEL MICROCONTROLADOR	30
2.2.1.7 MEMORIA DE DATOS (RAM)	32
2.2.1.7.1 RESUMEN DE ALGUNOS DE LOS REGISTROS DE CONFIGURACIÓN	33
2.2.2 TECNOLOGÍA GSM	34
2.2.2.1 ARQUITECTURA	34
2.2.2.2 DESARROLLO DE UNA LLAMADA GSM	37
2.2.2.3 VENTAJAS Y DESVENTAJAS ANTE OTRAS TECNOLOGÍA	38
2.2.2.3.1 VENTAJAS	38
2.2.2.3.2 DESVENTAJAS	38
2.2.2.4 LIMITACIONES DE GSM PARA LA TRANSMISIÓN DE DATOS	38
2.2.3 TECNOLOGÍA GPRS	39
2.2.3.1 ¿POR QUÉ SURGE GPRS?	39
2.2.3.2 ¿QUÉ ES GPRS?	39
2.2.3.3 ¿POR QUÉ ES MEJOR GPRS QUE GSM?	39
2.2.3.4 VENTAJAS DEL GPRS PARA EL USUARIO	40
2.2.3.5 SERVICIOS DEL GPRS PARA EL USUARIO	41
2.2.3.5.1 ACCEDER EN MOVILIDAD A INTERNET Y CORREO ELECTRÓNICO	41
2.2.3.5.2 ACCESO A CUENTAS DE CORREO CORPORATIVAS (INTRANET)	41
2.2.3.5.3 ACCESO A BASES DE DATOS Y APLICACIONES CORPORATIVAS DESDE UN DISPOSITIVO MÓVIL	41
2.2.3.5.4 ACCESO GPRS A APLICACIONES WAP PARA USO EMPRESARIALES (A TRAVÉS DEL SERVICIO WAP)	41

2.2.3.5.5 ACCESO A SERVICIOS DE INFORMACIÓN (A TRAVÉS DEL SERVICIO WAP)	41
2.2.3.6 VENTAJAS DEL GPRS PARA LA OPERADORA	42
2.2.3.7 ¿CÓMO SE ACCEDE A GPRS?	42
2.2.3.8 ¿CÓMO FUNCIONA GPRS?	42
2.2.3.9 MÓDULO GPRS/GSM (CELULAR) SIM900	45
2.2.3.10 ESPECIFICACIONES TÉCNICAS	45
2.2.3.11 DIAGRAMA FÍSICO DEL HARDWARE Y DESCRIPCIÓN	46
2.2.3.12 DIAGRAMA DE BLOQUES DEL MÓDULO SIM 900	47
2.2.3.13 EVALUACIÓN DEL SIM900	47
2.2.4 BLUETOOTH	48
2.2.4.1 HISTORIA DE BLUETOOTH	49
2.2.4.2 PRINCIPIOS DEL BLUETOOTH	49
2.2.4.3 VERSIONES DE BLUETOOTH	49
2.2.4.4 MÓDULO BLUETOOTH SLAVE (HC-06)	51
2.2.4.4.1 DESCRIPCIÓN GENERAL	51
2.2.4.4.2 ESPECIFICACIONES	52
2.2.5 MEDICIÓN DE LA EMISIÓN DE GASES EN VEHÍCULOS	52
2.2.5.1 DETECCIÓN DE GASES	52
2.2.5.2 MONÓXIDO DE CARBONO	53
2.2.5.3 MEDICIÓN DE LAS CONCENTRACIONES DE UN GAS	53
2.2.5.3.1 PARTES POR MILLÓN	53
2.2.5.4 CLASIFICACIÓN DE ESTADOS DE LA CALIDAD DEL AIRE	54
2.2.5.5 CLASIFICACIÓN DE LOS ESTADOS DE LA CALIDAD DEL AIRE	54
2.2.5.6 SENSORES DE GAS	54
2.2.5.6.1 SENSOR DE GAS MQ-2	55
2.2.5.6.2 CARACTERÍSTICAS	55
2.2.5.6.3 ESPECIFICACIONES	55
2.2.5.6.3.1 CONDICIONES DE TRABAJO	55
2.2.5.6.3.2 CONDICIÓN AMBIENTAL	56
2.2.5.6.3.3 CARACTERÍSTICAS DE SENSIBILIDAD	56

2.2.6. SISTEMA OPERATIVO ANDROID	57
2.2.6.1 ¿QUÉ ES ANDROID?	57
2.2.6.2 HISTORIA DE ANDROID	58
2.2.6.3 VERSIONES DISPONIBLES	58
2.2.6.2 ARQUITECTURA DE LA PLATAFORMA ANDROID	59
2.2.6.2.1 KERNEL	60
2.2.6.3 GUIA PARA DESARROLLADORES (BÁSICA)	60
2.2.6.4 PAUTAS PARA LAS INTERFACES DE USUARIO	62
2.2.6.5 APP INVENTOR	63
2.2.6.5.1 ¿PORQUÉ APP INVENTOR?	64
2.2.6.5.2 REQUERIMIENTOS DEL SISTEMA	64

CAPÍTULO III

3.0 CONSTRUCCIÓN DE LA APLICACIÓN	66
3.1 ESTUDIO DE FACTIBILIDAD	66
3.2 ANÁLISIS DEL SISTEMA	66
3.2.1 DESCRIPCIÓN DE LAS ACTIVIDADES	66
3.3 DISEÑO DEL SISTEMA	66
3.3.2 DISEÑO DEL HARDWARE DEL SISTEMA	66
3.3.2.1 DIAGRAMA DE BLOQUES DEL SISTEMA DISEÑADO	66
3.3.2.2 ESQUEMA DEL CIRCUITO EN EAGLE	67
3.3.3 DISEÑO DEL SOFTWARE DEL SISTEMA	69
3.3.3.1 PROGRAMACIÓN EN PIC BASIC PRO DEL MICROCON- TROLADOR PIC 16F877A	69
3.3.4 SOFTWARE DEL MEDIDOR DE CO PARA DISPOSITIVO MÓVIL	77
3.3.4.1 ESPECIFICACIONES DE LA APLICACIÓN PARA EL DISPOSITIVO MÓVIL	77
3.3.4.2 DESARROLLO DE APLICATIVO EN APPINVENTOR	78
3.3.4.2.1 COMPONENTES VISIBLES	79

3.3.4.2.2 COMPONENTES NO VISIBLES	80
3.3.4.3 DESCRIPCIÓN DEL PROGRAMA	81
3.3.4.3.1 VARIABLES	81
3.3.4.3.2 BOTONES	81
3.3.4.3.3 DESLIZADOR	84
3.3.4.3.4 TEMPORIZADOR	84
3.3.4.3.5 NOTIFICACIÓN	86

CAPÍTULO IV

4.0 ANÁLISIS E INTERPRETACIÓN DE RESULTADOS	87
4.1 PRUEBAS Y RESULTADOS AL PROTOTIPO	87
4.2 CONTRASTACIÓN DE HIPÓTESIS	91

CAPÍTULO V

5.0 CONCLUSIONES Y RECOMENDACIONES	92
5.1 CONCLUSIONES	92
5.2 RECOMENDACIONES	92
6.0 BIBLIOGRAFIA	93
7.0 ANEXOS	95

ÍNDICE DE TABLAS	PÁG
TABLA 2.0 CAPACIDAD MÁXIMA DE CORRIENTE TOTAL DE LOS PUERTOS DEL MICROCONTROLADOR	28
TABLA 2.1 DESCRIPCIÓN DE PINES DEL MICROCONTROLADOR 16F877A	29
TABLA 2.2 CONTINUACIÓN DE LA DESCRIPCIÓN DE PINES DEL MICROCONTROLADOR 16F877A	30
TABLA 2.3 CANALES DE CONTROL COMÚN UTILIZADOS POR LA ESTACIÓN BASE	35
TABLA 2.4 CANALES DE CONTROL DEDICADOS UTILIZADOS POR LA ESTACIÓN BASE	35
TABLA 2.5 CLASIFICACIÓN DE LOS ESTADOS DE LA CALIDAD DEL AIRE	54
TABLA 2.6 CONDICIONES DE TRABAJO DEL SENSOR MQ2	55
TABLA 2.7 CONDICIÓN AMBIENTAL DEL SENSOR MQ2	56
TABLA 2.8 CARACTERÍSTICAS DE SENSIBILIDAD	56
TABLA 3.0 DESCRIPCIÓN DE ACTIVIDADES	66

ÍNDICE DE FIGURAS	PÁG.
FIG. 2.0 ARQUITECTURA VON NEUMANN	23
FIG. 2.1 ARQUITECTURA HARVARD	24
FIG. 2.2 DISTRIBUCIÓN DE PINES DEL PIC16F877A	27
FIG. 2.3 FOTOGRAFIA DEL PIC 16F877A	28
FIG. 2.4 DIAGRAMA DE BLOQUES DEL MICROCONTROLADOR 16F877A	31
FIG. 2.5 ORGANIZACIÓN DE LA MEMORIA RAM DEL PIC16F877	32
FIG. 2.6 ARQUITECTURA DEL SISTEMA GSM	36
FIG. 2.7 EJEMPLO DE UN ENRUTAMIENTO DE LLAMADAS GSM	37
FIG. 2.8 PILA DE PROTOCOLOS DEL PLANO DE SEÑALIZACIÓN	44
FIG. 2.9 PROTOCOLOS GPRS	44
FIG. 2.10 MODULO GPRS/GSM SIM 900	45
FIG. 2.11 DIAGRAMA FISICO DEL MODULO SIM 900	46
FIG. 2.12 DIAGRAMA LOGICO DEL SIM 900	47
FIG. 2.13 FOTOGRAFIA DEL MODULO SIM 900	48
FIG. 2.14 SIMBOLO DE BLUETOOTH	48
FIG. 2.15 MODULO BLUETOOTH	51
FIG. 2.16 SENSOR DE GAS MQ-2	55
FIG. 2.17 DESCRIPCION DE CADA PIN	57
FIG. 2.18 SISTEMAS DE CAPAS DE ANDROID	59
FIG. 2.19 ENTORNO DE DESARROLLO DE APP INVENTOR	63
FIG. 2.20 INTERFAZ DE APP INVENTOR 2	65
FIG. 2.21 USO DE EMULADOR VIRTUAL PARA PRUEBA DE LA APLICACIÓN	65
FIG. 3.0 DIAGRAMA DE BLOQUES DEL SISTEMA	67
FIG. 3.1 CONEXIONES DEL SISTEMA IMPLEMENTADO	67
FIG. 3.2 RUTEO DEL CIRCUITO	68
FIG. 3.3 VISUALIZACION DE PARTE DEL PROGRAMA EN LA VENTANA DEL PBP	76
FIG. 3.4 PANTALLA PRINCIPAL DE LA APLICACIÓN EN DISPOSITIVO MÓVIL PARA MEDICIÓN DE CO	77
FIG. 3.5 PANTALLA DE DISEÑO DE APLICACIÓN EN APPINVENTOR2	78

FIG. 3.6 COMPONENTES VISIBLES DE LA APLICACIÓN	79
FIG. 3.7 COMPONENTES NO VISIBLES	80
FIG. 3.8 NOMBRES DE LOS DIFERENTES COMPONENTES DE LA APLICACIÓN	80
FIG 3.9 DISEÑO DE DECLARACIÓN DE VARIABLES DEL EDITOR DE BLOQUES DE APP INVENTOR2	81
FIG. 3.10 DISEÑO DEL BOTÓN CONECTAR DEL EDITOR DE BLOQUES DE APP INVENTOR2	81
FIG. 3.11 LISTPICKER (BEFOREPICKING)	82
FIG. 3.12 DISEÑO DEL BOTÓN LISTPICKER1	82
FIG 3.13 DISEÑO DEL BOTÓN SALIR	82
FIG. 3.14 DISEÑO DEL BOTÓN PARAR	83
FIG. 3.15 DISEÑO DEL BOTÓN SALVAR	83
FIG. 3.16 DISEÑO DEL BOTÓN REDBUTTON	83
FIG 3.17 DISEÑO DEL BOTÓN CLEARBUTTON	83
FIG. 3.18 VARIACIÓN DE LÍNEA DE REFERENCIA	84
FIG. 3.19 DISEÑO DEL CLOCK1 DEL EDITOR DE BLOQUES DE APPINVENTOR	84
FIG. 3.20 DISEÑO DEL CLOCK2	85
FIG. 3.21 DISEÑO DE NOTIFICACIÓN	86
FIG. 3.22 DISEÑO DE TEXTING1 (ENVIAR MENSAJE DE TEXTO)	86
FIG. 4.0 PROGRAMA VISUALIZADO EN APP INVENTOR	87
FIG. 4.1 APLICACIÓN VISUALIZADA EN UNA TABLET	87
FIG. 4.2 MEDICIÓN DEL HUMO DE CIGARRO	88
FIG. 4.3 MEDICIÓN VISUALIZADA EN UN CELULAR	88
FIG. 4.4 GRAFICA ADQUIRIDA DE LOS VALORES DEL SENSOR	89
FIG. 4.5 MEDIDA DE CO EN UN AUTO NISSAN	89
FIG. 4.6 MEDICIÓN DE CO EN UN AUTO SUZUKI	90
FIG. 4.7 VISUALIZACIÓN DE LA MEDIDA DEL CO EN UNA TABLET	90

ANEXOS

PÁG.

DATASHEET DEL PIC 16F877A	95
DATASHEET DEL SENSOR MQ-2	104
DATASHEET DEL MÓDULO BLUETOOTH SLAVE	108
DATASHEET DEL MÓDULO GSM SIM900	113

RESUMEN

La presente tesis realiza el desarrollo de un sistema de monitoreo de contaminación ambiental producido por los vehículos de transporte público. Así mismo hace énfasis en la necesidad de contar con un sistema que nos permita monitorizar los gases que emiten los motores durante la combustión al ambiente y estén dentro del rango de valores asequibles para evitar daño a los seres humanos.

En el primer capítulo se describe la problemática y se establece el objetivo general y los objetivos específicos de la tesis, así como la justificación, los alcances y los límites de la misma. El segundo capítulo hace referencia al marco teórico y las definiciones que nos permiten entender desde la base la aplicación desarrollada.

En el tercer capítulo se muestra el desarrollo de la aplicación paso a paso tanto en el diseño del hardware como del software.

El cuarto capítulo nos permite analizar los resultados obtenidos al probar la aplicación

En el quinto capítulo se presentan las conclusiones a las que se llegaron con el desarrollo de la aplicación; y se enuncian las recomendaciones a seguir en el desarrollo de la aplicación.

PALABRAS CLAVE:

PBP (PIC BASIC PRO)

PIC (CONTROLADOR DE INTERFAZ PERIFÉRICO)

CO (MONÓXIDO DE CARBONO)

SIG (SPECIAL INTEREST GROUP)

ABSTRACT

The present thesis is accomplishes the development of a system of monitoring of environmental contamination produced by the public wagons. Likewise emphasis in the need makes of counting on a system that monitorizar allow us to the gases that emit the motors during the combustion to the environment and be within the range of accessible moral values to avoid damage to the human beings.

The problems is described in the first chapter and the general objective and the specific objectives of the thesis, as well as justification, are established catch up with them and the limits of the same. The second chapter makes reference to the theoretic frame and the definitions that they allow us to understanding from the base developed application.

The development of application is shown step by step so much in the hardware's design in the third chapter as if from the software.

The fourth chapter allows us examining the results obtained when trying the application. They encounter the findings they took place to with the development of the application in the fifth chapter; And they enunciate recommendations to follow in the development of application.

KEY WORDS

PBP (PIC BASIC PRO)

PIC (PERIPHERAL INTERFACE CONTROLLER)

CO (CARBON MONOXIDE)

SIG (GRUPO DE INTERESES ESPECIALES)

INTRODUCCIÓN

La presente investigación se refiere al tema de monitoreo de la contaminación ambiental por los vehículos de transporte público ya que un sistema de contaminación del aire es un grave problema de salud ambiental que afecta a países desarrollados y en desarrollo de todo el mundo, en una escala global, se emiten a la atmósfera grandes cantidades de partículas y gases potencialmente nocivos que afectan al ambiente, nuestro ecosistema y desde luego la salud de todos los seres humanos que habitamos el planeta.

La característica principal es el crecimiento incesante de las grandes industrias, el aumento del parque automotor son algunas de las fuentes de contaminación que contribuyen al calentamiento global. El sector transporte en el Perú utiliza una gran cantidad de combustible fósil, principalmente petróleo diésel y gasolina. En los últimos años el crecimiento del consumo de petróleo diésel se ha incrementado velozmente, mientras que el consumo de gasolina se ha mantenido casi estable.

La contaminación por combustión es un problema latente en la actualidad, anualmente diversas empresas realizan monitoreo de emisiones dando muestra a una elevada contaminación del aire. En años recientes el mundo de la electrónica y en especial el diseño de sistemas embebidos basados en microcontroladores nos permiten desarrollar toda una gama de aplicaciones para monitorizar cualquier variable que se requiera.

El sistema diseñado y explicado en este trabajo monitoriza la emisión de monóxido de carbono emitido por los vehículos de transporte público, dicho sistema nos permite visualizar los valores permisibles de CO durante la combustión de los motores permitiéndonos evaluar el estado del buen funcionamiento de los mismos.

El sistema consta de un microcontrolador PIC 16F877A, un sensor de CO MQ-2, un módulo bluetooth y una tarjeta GSM SIM900; la interface y visualización de la aplicación se realiza a través de un celular o tablet. Se propone una solución con ayuda de algunas tecnologías, y software tales como el software App inventor2, el software Pic Basic Pro, un equipo celular con sistema operativo android.

La aplicación puede ser utilizada por personas en general, aficionados o empresas dedicadas a la realización de controles de calidad en el parque automotor.

CAPÍTULO I

PLANTEAMIENTO METODOLÓGICO

1.0 DESCRIPCIÓN DE LA REALIDAD PROBLEMÁTICA

La contaminación del aire es una problemática de carácter mundial, su estudio es de gran importancia por las consecuencias que tiene tanto en el medio ambiente como en la salud de los seres vivos que lo conforman.

Los vehículos automotores son una de las principales fuentes de contaminantes atmosféricos, son propulsados por motores de combustión interna (gasolina, diesel o algún otro derivado del petróleo).

Debido a que la combustión no es perfecta ello genera emisiones contaminantes por el tubo de escape: CO (Monóxido de carbono), NO₂ (óxido nitroso), SO₂ (dióxido de azufre), HC (hidrocarburos), Pb (caso gasolinas con plomo), y CO₂ (dióxido de carbono), CH₄ (metano), N₂O (óxido nitroso) (vehículos diesel). Con la realización del presente trabajo se pretende realizar mediciones que permitan cuantificar la cantidad de CO que es emitida a la atmósfera por un vehículo, un cigarro, en la cocina de una casa/restaurante, etc. ya que esto contribuyen en gran medida a la contaminación del aire.

1.1 DELIMITACIONES Y DEFINICIÓN DEL PROBLEMA.

1.1.1 DELIMITACIONES

El presente trabajo se plantea implementar un Sistema de Monitoreo de la contaminación Ambiental producida por los vehículos de transporte Público. Para tal efecto se plantea diseñar un equipo electrónico que se instalara en cada vehículo y este pueda realizar las mediciones de emisión de Monóxido de Carbono y entre otros gases que contribuyan más a la contaminación. Los valores obtenidos se enviara vía Mensajes de texto usando tecnología GSM o vía datos utilizando tecnología GPRS hacia una central de control Ambiental, la cual determinara las acciones de control respectiva con respecto al vehículo según los valores obtenidos.

1.1.2 DEFINICIÓN DEL PROBLEMA

La contaminación del aire es un importante problema de salud ambiental que afecta a países desarrollados y en desarrollo de todo el mundo, en una escala global, se emiten a la atmósfera grandes cantidades de partículas y gases potencialmente nocivos que afectan la salud humana y el ambiente y que en el largo plazo dañan los recursos necesarios para el desarrollo sostenible del planeta. El crecimiento incesante de las grandes industrias, el aumento del parque automotor son algunas de las fuentes de contaminación que contribuyen al calentamiento global.

El sector transporte en el Perú utiliza una gran cantidad de combustible fósil, principalmente petróleo diesel y gasolina. La contaminación por combustión es un problema latente en la actualidad, anualmente diversas empresas realizan monitoreo de emisiones dando muestra a una elevada contaminación del aire.

1.2 FORMULACIÓN DEL PROBLEMA

1.2.1 PROBLEMA PRINCIPAL

La contaminación ambiental debido a los gases emitidos por los motores de los vehículos de transporte públicos ha hecho en las grandes ciudades del mundo la disminución de la calidad del aire respirable en el ambiente, y por ello afluencia de muchas enfermedades respiratorias en los seres humanos. La gran demanda de automóviles en las metrópolis y la incesante demanda en el uso de combustible hacen necesario e imprescindible tener y hacer valer los reglamentos emitidos por la OMS respecto al valor de emisión de gases por dichos vehículos al ambiente y que estos permanezcan en el rango adecuado. Por ello se propone diseñar un sistema para medir los valores de emisiones de monóxido de carbono y así evaluar el control de calidad en los motores.

1.3 OBJETIVOS DE LA INVESTIGACIÓN

1.3.1 OBJETIVO GENERAL

Diseñar un Sistema de Monitoreo de la contaminación Ambiental, producida por los vehículos de transporte público.

1.3.2 OBJETIVOS ESPECÍFICOS

- Diseñar sistema de detección y/o medición de emisión de monóxido de carbono y otros gases que contribuyan a la contaminación ambiental.
- Utilizar el módulo de comunicación GSM SIM900 para envío de mensajes.
- Diseñar software para dispositivo móvil
- Diseñar software para PC de escritorio

1.4 HIPÓTESIS GENERAL

1.4.1 HIPÓTESIS GENERAL

Usando las tecnologías existentes de sensores, microcontroladores, y demás circuitos integrados así como seleccionando el software adecuado si es posible Diseñar un Sistema de Monitoreo de contaminación Ambiental, producida por vehículos de transporte público.

CAPITULO II

2.0 MARCO TEÓRICO

2.1 ANTECEDENTES DE LA INVESTIGACIÓN

Los estudios consultados, que guardan relación con la temática abordada en este trabajo se describen a continuación:

- En el año 2013, se desarrolló un Trabajo de Grado para optar el título de grado, en la Universidad Nacional del Sur, específicamente en el departamento de Ingeniería Eléctrica y de Computadoras titulado: “Desarrollo e Implementación de un Sistema Embebido: Nariz Electrónica”, el objetivo planteado fue diseñar un sistema para la detección y medición de gases como el metano/butano, entre otros agentes. Este sistema se basó en un microcontrolador de la firma ATMEL, sensores de gas de la serie MQ, pulsadores, una pantalla LCD para brindar la información al usuario.
- En el año 2011, se desarrolló un Trabajo de grado para optar el título de grado, en la Universidad Carlos III de Madrid, específicamente en el departamento de Ingeniería de Sistemas y Automática titulado: “Desarrollo Placa de Sensores para Monitorización de Gases Tóxicos”, el objetivo de trabajo fue desarrollar un sistema de sensorización y monitorización continua de la calidad del aire, para ello eligieron los sensores más adecuados para esta aplicación además de un sistema de alerta capaz de detectar de forma temprana y eficiente concentraciones nocivas de gases para tomar las medidas oportunas en caso de contaminación del aire.
- En el año 2010, se desarrolló un Trabajo de Grado para optar el título de grado, en la Universidad Nueva Esparta, específicamente en la escuela de Electrónica, titulado: “Desarrollo de un Prototipo para la Detección de Fugas de Gas en Tuberías”, el objetivo planteado fue diseñar un prototipo para la detección de fugas de gas, compuesto por un módulo sensor de gas, un sistema digital de visualización, un sistema de alarmas. Se efectuaron pruebas de laboratorio, obteniendo los datos necesarios para la programación del equipo. La investigación de los autores, representa un antecedente importante para este estudio; por cuanto proporciona información sobre los procedimientos que deben seguir en el diseño del circuito electrónico y la respectiva programación.
- En el año 2007 dos bachilleres efectuaron un Trabajo de Grado para optar al título de grado, en la Universidad Nueva de Esparta, específicamente en la Escuela de Electrónica, titulado: “Diseño e Implementación de un Sistema de Detección de Gas Licuado de Petróleo (GLP) para uso doméstico”. En este estudio los autores se plantearon como objetivo general realizar el diseño de un sistema capaz de detectar la presencia de gas licuado de petróleo en ambientes domésticos donde existen u operen artefactos que usen este tipo de gas, con la finalidad de evitar accidentes producidos por fugas lo cual ocasionaría consecuencias negativas como incendios y algunas pérdidas humanas.

Esta serie de investigaciones se han considerado como antecedentes por cuanto aportan información significativa y relevante, sobre los procesos que se deben considerar en la construcción del circuito electrónico para detectar y monitorear CO (monóxido de carbono), a través de la programación de un microcontrolador y un sensor de gas.

2.2 MARCO CONCEPTUAL

2.2.1 MICROCONTROLADORES

2.2.1.1 ¿QUÉ ES UN MICROCONTROLADOR?

El microcontrolador es un circuito integrado de muy alta escala de integración que contiene las partes funcionales de un computador:

- CPU (Central Processor Unit o Unidad de Procesamiento Central)
- Memorias volátiles (RAM), para datos
- Memorias no volátiles (ROM, PROM, EPROM) para escribir el programa
- Líneas de entrada y salida para comunicarse con el mundo exterior.
- Algunos periféricos (comunicación serial, temporizador, convertidor A/D, etc.)

“Es decir el microcontrolador es un computador integrado en un solo chip. Integrar todos estos elementos en un solo circuito integrado ha significado desarrollar aplicaciones importantes en la industria al economizar materiales, tiempo y espacio”. [1]

2.2.1.2 ¿QUE PUEDO HACER CON EL MICROCONTROLADOR?

Las aplicaciones de un microcontrolador son tan inmensas que el límite es la propia imaginación del usuario. Estos microcontroladores están en el auto, en el televisor, en el teléfono, en una impresora, en un horno de microondas, en un juguete, en un transbordador espacial etc.

Los siguientes son algunos campos en los que los microcontroladores tienen gran uso:

- En la industria del automóvil: Control de motor, alarmas, regulador del servofreno, dosificador, etc.
- En la industria de los electrodomésticos: control de calefacciones, lavadoras, cocinas eléctricas, etc.
- En informática: como controlador de periféricos. Controlar impresoras, plotters, cámaras, scanners, terminales, unidades de disco, teclados, módems, etc.
- En la industria de imagen y sonido: tratamiento de la imagen y sonido, control de los motores de arrastre del giradiscos, magnetófono, video, etc.
- **En la industria, en general se utilizan en:**
 - Regulación: todas las familias de microcontroladores incorporan en alguna de sus versiones conversores A/D y D/A, para la regulación de la velocidad de las máquinas, de niveles, de temperatura, etc.
 - Automatismos: La enorme cantidad de líneas de entrada y salidas, y su inmunidad al ruido le hacen muy valioso para el control secuencial de procesos. Por ejemplo control de máquinas, herramientas, apertura y cierre automático de puertas según condiciones, plantas empaquetadoras, aparatos de maniobra de ascensores, etc.
 - Robótica: para control de los motores y captura de señales de los diferentes sensores, fabricación de controladores robóticos para sistemas automáticos, etc.

[1] <http://www.parallax.com/education>

- **Instrumentos portátiles compactos:**
 - Radio paginador numérico (beeper)
 - Planímetro electrónico
 - Nivelímetro digital
 - Identificador-probador de circuitos integrados
 - Tacómetro digital
 - Panel frontal de un osciloscopio
 - Controlador de display LCD
 - Analizador de espectros, etc.
- **Dispositivos autónomos:**
 - Fotocopiadoras
 - Máquinas de escribir
 - Selector, Codificador decodificador de TV
 - Localizador de peces
 - Teléfonos de tarjeta
 - Teléfonos celulares
 - Cerraduras electrónicas
 - Sistemas de seguridad

Se emplea también en medicina, en aplicaciones militares, edificios inteligentes, etc.

2.2.1.2.1 PRINCIPALES FABRICANTES

Por lo general los fabricantes de microprocesadores, lo son de microcontroladores. Los fabricantes de microcontroladores son más de 50, podemos mencionar a:

- ATMEL
- MOTOROLA
- INTEL
- MICROCHIP
- NEC
- HITACHI
- MITSUBISHI
- PHILIPS
- MATSUSHITA
- TOSHIBA
- AT&T
- ZILOG
- SIEMENS
- NATIONAL SEMICONDUCTOR, ETC. [2]

[2] <http://doru.info/projects/hdl/pavr/index.html>

2.2.1.3 APARICIÓN Y DESARROLLO DE LOS MICROCONTROLADORES

2.2.1.3.1 BREVE ESBOZO HISTÓRICO

La siguiente es una lista cronológica de los eventos tecnológicos más recientes que han tenido impacto sobre la aparición y el desarrollo del campo de los microcontroladores en la electrónica digital.

1971: Intel fabrica el primer microprocesador (el 4004) de tecnología PMOS. Este era un microprocesador de 4 bits y fue fabricado por Intel a petición de Datapoint Corporation con el objeto de sustituir la CPU de terminales inteligentes fabricadas en esa fecha por Datapoint mediante circuitería discreta. El dispositivo fabricado por Intel resultó 10 veces más lento de lo requerido y Datapoint no lo compró, de esta manera Intel comenzó a comercializarlo. El 4004 podía direccionar sólo 4096 (4k) posiciones de memoria de 4 bits, reconocía 45 instrucciones y podía ejecutar una instrucción en 20 μ seg en promedio.

1972: Las aplicaciones del 4004 estaban muy limitadas por su reducida capacidad y rápidamente Intel desarrolló una versión más poderosa (el 8008), el cual podía manipular bytes completos, por lo cual fue un microprocesador de 8 bits. La memoria que este podía manejar se incrementó a 16 kbytes, sin embargo, la velocidad de operación continuó igual.

1973: Intel lanza al mercado el 8080 el primer microprocesador de tecnología NMOS, lo cual permite superar la velocidad de su predecesor (el 8008) por un factor de diez, es decir, el 8080 puede realizar 500K de operaciones por segundo, además se incrementó la capacidad de direccionamiento de memoria a 64 kbytes. A partir del 8080 de Intel se produjo una revolución en el diseño de microcomputadoras y varias compañías fabricantes de circuitos integrados comenzaron a producir microprocesadores. Algunos ejemplos de los primeros microprocesadores son: el IMP-4 y el SC/MP de National Semiconductors, el PPS-4 y PPS-8 de Rockwell International, el MC6800 de Motorola, el F-8 de Fairchild.

1975: Zilog lanza al mercado el Z80, uno de los microprocesadores de 8 bits más poderosos. En ese mismo año, Motorola reduce sus costos con sus microprocesadores 6501 y 6502 (este último adoptado por APPLE para su primera microcomputadora personal). Estos microprocesadores se comercializan en \$20 y \$25 (DLS USA) respectivamente. Esto provoca un auge en el mercado de microcomputadoras de uso doméstico y un caos en la proliferación de lenguajes, sistemas operativos y programas (ningún producto era compatible con el de otro fabricante).

1976: Surgen las primeras microcomputadoras de un solo chip, que más tarde se denominarán microcontroladores. Dos de los primeros microcontroladores, son el 8048 de Intel y el 6805R2 de Motorola.

198x: En la década de los 80's comienza la ruptura entre la evolución tecnológica de los microprocesadores y la de los microcontroladores, Ya que los primeros han ido incorporando cada vez más y mejores capacidades para las aplicaciones en donde se requiere el manejo de grandes volúmenes de información y por otro lado, los segundos han incorporado más capacidades que les permiten la interacción con el mundo físico en tiempo real, además de mejores desempeños en ambientes de tipo industrial.

2.2.1.4 ARQUITECTURA DE UN MICROCONTROLADOR

Según la arquitectura interna de la memoria de un microcontrolador se puede clasificar considerando como el CPU accede a los datos e instrucciones, en 2 tipos: [3]

2.2.1.4.1 ARQUITECTURA VON NEUMANN

Fue desarrollada por Jon Von Neumann, se caracteriza por tener una sola memoria principal donde se almacenan datos e instrucciones de forma indistinta. La CPU se conecta a través de un sistema de buses (direcciones, datos y control). Esta arquitectura es limitada cuando se demanda rapidez.

Memoria

Bus de direcciones Instrucciones

CPU +

Datos

Bus de datos

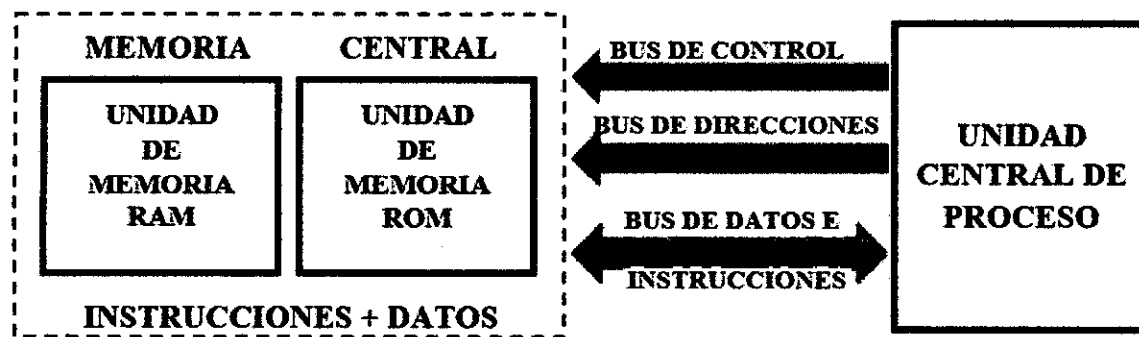


FIG. 2.0 ARQUITECTURA VON NEUMANN

2.2.1.4.2 ARQUITECTURA HARVARD

Fue desarrollado en Harvard, por Howard Aiken, esta arquitectura se caracteriza por tener 2 memorias independientes una que contiene sólo instrucciones y otra, que contiene sólo datos. Ambas, disponen de sus respectivos sistemas de buses para el acceso y es posible realizar operaciones de acceso simultáneamente en ambas memorias.

Existe una variante de esta arquitectura que permite el acceso a la tabla de datos desde la memoria de programas, es la Arquitectura de Harvard Modificada.

Esta última arquitectura es la dominante en los microcontroladores actuales ya que la memoria de programas es usualmente ROM, OTP, EPROM o FLASH, mientras que la memoria de datos es usualmente RAM.

Por ejemplo las tablas de datos pueden estar en la memoria de programa sin que sean perdidas cada vez que el sistema es apagado.

[3] Microcontroladores Pic, programación en basic. Carlos A. Reyes, 2da edic. pág. 18

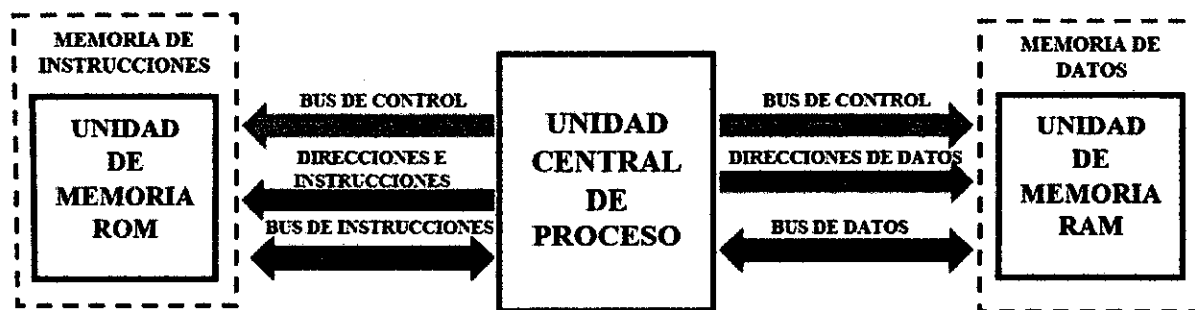


FIG. 2.1 ARQUITECTURA HARVARD

Se puede observar claramente que las principales ventajas de esta arquitectura son:

- a) El tamaño de las instrucciones no está relacionado con el de los datos, y por lo tanto puede ser optimizado para que cualquier instrucción ocupe una sola posición de memoria de programa, logrando así mayor velocidad y menor longitud de programa.
- b) El tiempo de acceso a las instrucciones puede superponerse con el de los datos, logrando una mayor velocidad de operación.

2.2.1.5 MICROCONTROLADOR A ELEGIR

2.2.1.5.1 ¿QUÉ MICROCONTROLADOR ELEGIR?

Sin duda la elección del microcontrolador dependerá de la tarea o proyecto que se tiene en mente pues los fabricantes como se mencionó anteriormente son más de 50, estos tienen muchos modelos enfocados a tareas específicas. Esta selección deberá ir de la mano con factores económicos óptimos así como de la idea del controlador incrustado (embedded controller), el cual es un controlador dedicado a una sola tarea e incorporado al sistema que gobierna.

- **PROCESAMIENTO DE DATOS:** Cuando se desea realizar cálculos complejos en un tiempo limitado, se debe seleccionar un microcontrolador suficientemente rápido para ello. Por otro lado, habrá que tener en cuenta la precisión de los datos a manejar: si no es suficiente con un microcontrolador de 8 bits, puede ser necesario acudir a microcontroladores de 16 ó 32 bits, o incluso a hardware de coma flotante.
- **ENTRADA/SALIDA:** Se debe identificar la cantidad y tipo de señales a controlar. Una vez realizado este análisis puede ser necesario añadir periféricos externos o cambiar a otro microcontrolador más adecuado a ese sistema.
- **CONSUMO:** algunos productos que incorporan microcontroladores están alimentados con baterías, puede ser que el microcontrolador esté trabajando en estado de bajo consumo pero debe “despertar” ante la activación de una señal (por ejemplo una interrupción) y ejecutar el programa adecuadamente.

- **MEMORIA:** para detectar las necesidades de memoria de una aplicación debemos saber la cantidad y el tipo de memoria necesaria, para esto se debe tener una versión preliminar (pseudo-código) de la aplicación y escoger el microcontrolador apropiado.

- **ANCHO DE PALABRA:** el criterio de diseño debe ser seleccionar el microcontrolador de menor ancho de palabra que satisfaga los requerimientos de la aplicación. Usar un microcontrolador de 4 bits supondrá reducir los costos, mientras que uno de 8 bits puede ser el más adecuado si el ancho de los datos es de un byte. Los microcontroladores de 16 y 32 bits, debido a su elevado costo, deben reservarse para aplicaciones que requieran altas prestaciones (Entrada/Salida grande o espacio de direccionamiento muy elevado).

- **DISEÑO DE LA PLACA:** la selección de un microcontrolador concreto condicionará el diseño de la placa de circuitos impresos.

2.2.1.6 EL MICROCONTROLADOR 16F877A

El microcontrolador PIC16F877 de Microchip pertenece a una gran familia de microcontroladores de 8 bits (bus de datos) que tienen las siguientes características generales que los distinguen de otras familias:

- Arquitectura Harvard
- Tecnología RISC
- Tecnología CMOS

Estas características se conjugan para lograr un dispositivo altamente eficiente en el uso de la memoria de datos y programa y por lo tanto en la velocidad de ejecución.

El PIC16F877 es un microcontrolador con memoria de programa tipo FLASH, lo que representa gran facilidad en el desarrollo de prototipos y en su aprendizaje ya que no se requiere borrarlo con luz ultravioleta como las versiones EPROM, sino que permite reprogramarlo nuevamente sin ser borrado con anterioridad.

“El PIC16F877 es un microcontrolador de tecnología Microchip fabricado en tecnología CMOS, su consumo de potencia es muy bajo y además es completamente estático, esto quiere decir que el reloj puede detenerse y los datos de la memoria no se pierden”. [4]

El encapsulado más común para este microcontrolador es el DIP (Dual In-line Pin) de 40 pines, propio para usarlo en experimentación. La referencia completa es PIC16F877-04 para el dispositivo que utiliza cristal oscilador de hasta 4 MHz, PIC16F877-20 para el dispositivo que utiliza cristal oscilador de hasta 20 MHz o PIC16F877A-I para el dispositivo tipo industrial que puede trabajar hasta a 20 MHz.

[4] <http://www.mikroe.com/chapters/view/84/libro>

Sin embargo, hay otros tipos de encapsulado que se pueden utilizar según el diseño y la aplicación que se quiere realizar. Por ejemplo, el encapsulado tipo surface mount (montaje superficial) tiene un reducido tamaño y bajo costo, que lo hace propio para producciones en serie o para utilizarlo en lugares de espacio muy reducido.

2.2.1.6.1 CARACTERÍSTICAS PRINCIPALES

- CPU RISC de alta performance
- Set de 35 instrucciones
Todas las instrucciones son de un ciclo salvo aquellas que incluyen saltos que son de 2 ciclos.
- Velocidad de Trabajo:
 - DC - 20 MHz de reloj de entrada
 - DC - 200 ns ciclo de instrucción
- Hasta 8K x 14 de trabajo y memoria FLASH de programación.
- Hasta 368 x 8 bytes, data de memoria (RAM)
- Hasta 256 x 8 bytes de Datos en Memoria EEPROM, manejo de Interrupciones (hasta 14 fuentes)
- Unidades aisladoras en hardware de 8 niveles
- Modo de direccionamiento directo, indirecto y relativo.
- Power-on Reset (POR)
- Power-up Timer (PWRT) y Oscilador Start-up Timer (OST)
- Watchdog Timer (WDT) con el reloj RC interno para mejor seguridad.
- Protección de código programable.
- Programación serial via 2 pines, programación serial en el circuito (ICSP)
- In-Circuit Debugging vía 2 pines
- Amplio rango de voltaje de trabajo: 2.0V a 5.5V

2.2.1.6.2 CARACTERÍSTICAS PERIFÉRICAS

- Timer0: 8-bit timer/counter con 8-bit de pre-escala
- Timer1: 16-bit timer/counter con pre-escala, que puede ser incrementado durante el modo SLEEP vía reloj externo.
- Timer2: 8-bit timer/counter con registro de periodo de 8-bit, prescaler y postscaler
- Dos módulos de captura, comparativa, PWM
- Captura de 16-bit, Máx. Resolución: 12.5 ns
- Compara 16-bit, máx. Resolución: 200 ns
- PWM máx. Resolución: 10-bit
- Convertidor Analógico - Digital de 10-bit multicanal
- Puerto de sincronización serial (SSP) con SPI (Modo maestro) e I2C (Maestro/Esclavo)
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) con detección de direcciones de 9-bit
- Puerto paralelo esclavo (PSP) de 8-bits de ancho, con controles externos de RD, WR y CS (solo 40/44-pin)

Brown-out detection circuitry para Brown-out Reset (BOR)

2.2.1.6.3 CONFIGURACIÓN DE PINES

Los pines de entrada/salida de este microcontrolador están organizados en cinco puertos, el puerto A con 6 líneas, el puerto B con 8 líneas, el puerto C con 8 líneas, el puerto D con 8 líneas y el puerto E con 3 líneas.

Cada pin de esos puertos se puede configurar como entrada o como salida independiente programando un par de registros diseñados para tal fin. En ese registro un bit en "0" configura el pin del puerto correspondiente como salida y un bit en "1" lo configura como entrada. Dichos pines del microcontrolador también pueden cumplir otras funciones especiales, siempre y cuando se configuren para ello, según se verá más adelante.

Los pines del puerto A y del puerto E pueden trabajar como entradas para el convertidor Análogo a Digital interno, es decir, allí se podría conectar una señal proveniente de un sensor o de un circuito analógico para que el microcontrolador la convierta en su equivalente digital y pueda realizar algún proceso de control o de instrumentación digital. El pin RB0/INT se puede configurar por software para que funcione como interrupción externa, para configurarlo se utilizan unos bits de los registros que controlan las interrupciones.

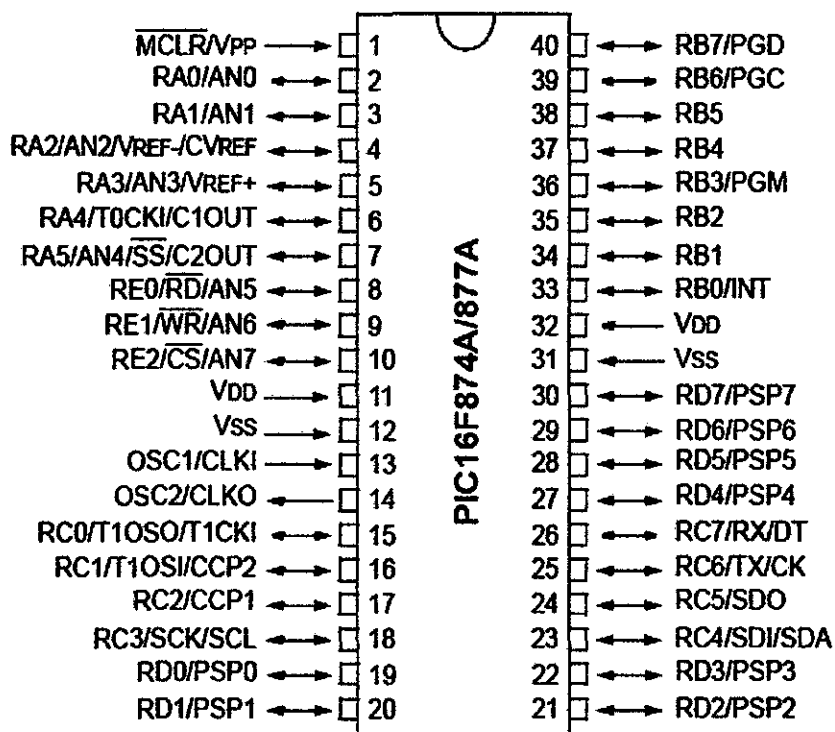


FIG. 2.2 DISTRIBUCIÓN DE PINES DEL PIC16F877A

El pin RA4/T0CKI del puerto A puede ser configurado como un pin de entrada/salida o como entrada del temporizador/contador. Cuando este pin se programa como entrada digital, funciona como un disparador de Schmitt (Schmitt trigger), puede reconocer señales un poco distorsionadas y llevarlas a niveles lógicos (cero y cinco voltios).

Cuando se usa como salida digital se comporta como colector abierto (open collector), por lo tanto, se debe poner una resistencia de pull-up (resistencia externa conectada a un nivel de cinco voltios).

Como salida, la lógica es inversa: un "0" escrito al pin del puerto entrega en el pin un "1" lógico. Además, como salida no puede manejar cargas como fuente, sólo en el modo sumidero.

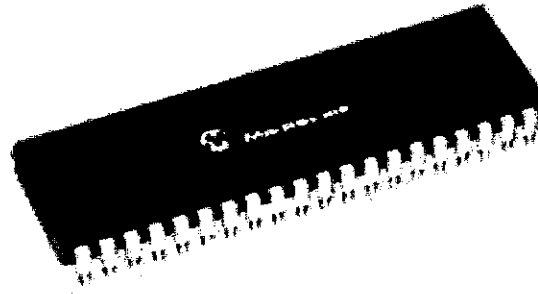


FIG. 2.3 FOTOGRAFIA DEL PIC 16F877A

	PUERTO A	PUERTO B	PUERTO C	PUERTO D
Modo sumidero	150 mA	200 mA	200 mA	200 mA
Modo fuente	150 mA	200 mA	200mA	200mA

TABLA 2.0 CAPACIDAD MÁXIMA DE CORRIENTE TOTAL DE LOS PUERTOS DEL MICROCONTROLADOR

El consumo de corriente del microcontrolador para su funcionamiento depende del voltaje de operación, la frecuencia y de las cargas que tengan sus pines. Para un oscilador de 4 MHz el consumo es de aproximadamente 2 mA; aunque este se puede reducir a 40 microamperios cuando se está en el modo sleep (en este modo el micro se detiene y disminuye el consumo de potencia).

2.2.1.6.4 DESCRIPCIÓN DE LOS PINES DEL MICROCONTROLADOR

NOMBRE PIN	PIN	DESCRIPCIÓN
RA0/AN0	2	E/S Digital o Entrada análoga 0.
RA1/AN1	3	E/S Digital o Entrada análoga 1.
RA2/AN2 Vref -	4	E/S Digital o Entrada análoga 2.
RA3/AN3/Vref +	5	E/S Digital o Entrada análoga 3.
RA4/T0CKI	6	Bit 4 del puerto A (E/S bidireccional). También se usa como entrada de reloj al temporizador/contador TMR0.
RA5/SS/AN4	7	E/S Digital o Entrada análoga 4. También lo usa el puerto serial síncrono.
RB0/INT	33	Bit 0 del puerto B (E/S bidireccional). Buffer E/S: TTL/ST. También se usa como entrada de interrupción externa (INT).
RB1	34	Bit 1 del puerto B (E/S bidireccional). Buffer E/S: TTL
RB2	35	Bit 2 del puerto B (E/S bidireccional). Buffer E/S: TTL
RB3/PGM	36	Bit 3 del puerto B (E/S bidireccional). Buffer E/S: TTL (Programación en bajo voltaje)
RB4	37	Bit 4 del puerto B (E/S bidireccional). Buffer E/S: TTL. Interrupción por cambio del pin.
RB5	38	Bit 5 del puerto B (E/S bidireccional). Buffer E/S: TTL. Interrupción por cambio del pin.
RB6/PGC	39	Bit 6 del puerto B (E/S bidireccional). Buffer E/S: TTL/ST. Interrupción por cambio del pin. Entrada de reloj para programación serial.
RB7/PGD	40	Bit 7 del puerto B (E/S bidireccional). Buffer E/S: TTL/ST. Entrada de datos para programación serial.
RC0/T1OSO/T1CKI	15	E/S Digital. Salida del oscilador Timer1 o entrada de reloj Timer1.
RC1/T1OSI/CCP2	16	E/S Digital. Entrada del oscilador Timer 1. Entrada Captura 2; Salida Compara 2; Salida PWM 2
RC2/CCP1	17	E/S Digital. Entrada Captura 1; Salida Compara 1; Salida PWM 1
RC3/SCK/SCL	18	E/S Digital. Línea de reloj serial asíncrono en el modo SPI y el modo I ² C
RC4/SDI/SDA	23	E/S Digital. Línea de datos en el modo SPI o en el modo I ² C
RC5/SDO	24	E/S Digital.
RC6/TX/CK	25	E/S Digital. Transmisión asíncrona (USART) o reloj síncrono (SSP).

TABLA 2.1 DESCRIPCIÓN DE PINES DEL MICROCONTROLADOR 16F877A

RC7/RX/DT	26	E/S Digital. Recepción asíncrona (USART) o línea de datos (SSP).
VDD	11,32	Voltaje de alimentación DC (+)
VSS	12,31	Referencia de voltaje (GND).
MCLR	1	Entrada de RESET al microcontrolador. Voltaje de entrada durante la programación. En nivel bajo resetea el microcontrolador.
OSC1/CLKIN	13	Entrada oscilador cristal oscilador / Entrada fuente de reloj externa.
OSC2/CLKOUT	14	Salida oscilador cristal. Oscilador RC: Salida con un ¼ frecuencia OSC1
RD0/PSP0	19	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD1/PSP1	20	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD2/PSP2	21	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD3/PSP3	22	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD4/PSP4	27	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD5/PSP5	28	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD6/PSP6	29	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD7/PSP7	30	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RE0/RD/AN5	8	E/S Digital. Puede ser pin de lectura (read) en modo microprocesador.
RE1/WR/AN6	9	E/S Digital. Puede ser pin de escritura (write) en modo microprocesador.
RE2/CS/AN7	10	E/S Digital. Puede ser pin de selección de chip (chip select) en modo microprocesador.

TABLA 2.2 CONTINUACIÓN DE LA DESCRIPCIÓN DE PINES DEL MICROCON 16F877A

2.2.1.6.5 ARQUITECTURA INTERNA DEL MICROCONTROLADOR

Este término se refiere a los bloques funcionales internos que conforman el microcontrolador y la forma en que están conectados, por ejemplo la memoria FLASH (de programa), la memoria RAM (de datos), los puertos, la lógica de control que permite que todo el conjunto funcione, etc.

La figura 2.4 muestra la arquitectura general del PIC16F877, en ella se pueden apreciar los diferentes bloques que lo componen y la forma en que se conectan. Se muestra la conexión de los puertos, las memorias de datos y de programa, los bloques especiales como el Watchdog, los temporizadores de arranque, el oscilador, etc.

Todos los elementos se conectan entre sí por medio de buses. Un bus es un conjunto de líneas que transportan información entre dos o más módulos. Vale la pena destacar que el PIC16F877 tiene un bloque especial de memoria de datos de 256 bytes del tipo EEPROM, además de los dos bloques de memoria principales que son el de programa y el de datos o registros.

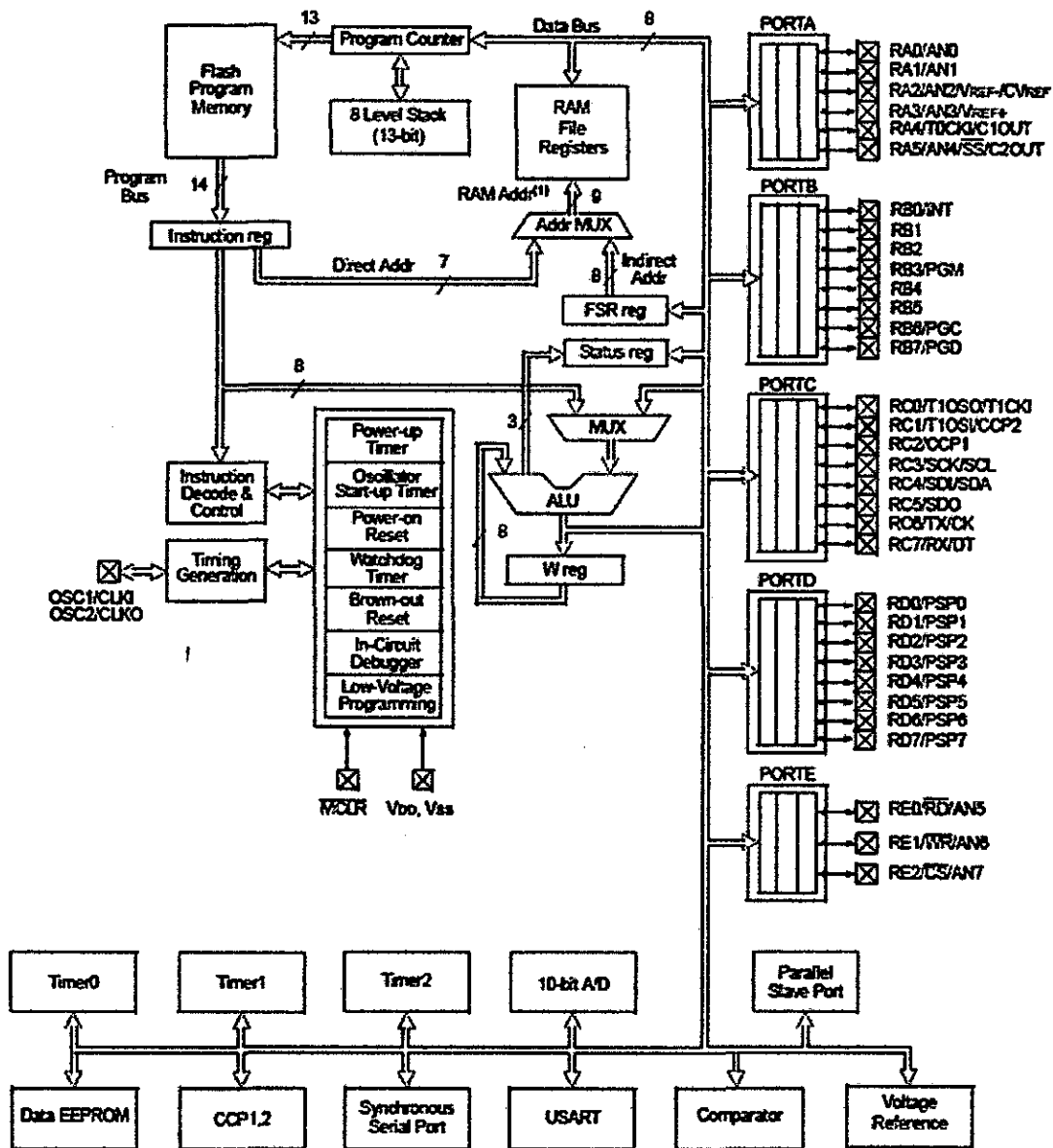


FIG. 2.4 DIAGRAMA DE BLOQUES DEL MICROCONTROLADOR 16F877A

El PIC16F877 se basa en la arquitectura Harvard, en la cual el programa y los datos se pueden trabajar con buses y memorias separadas, lo que posibilita que las instrucciones y los datos posean longitudes diferentes. Esta misma estructura es la que permite la superposición de los ciclos de búsqueda y ejecución de las instrucciones, lo cual se ve reflejado en una mayor velocidad del microcontrolador.

2.2.1.7 MEMORIA DE DATOS (RAM)

El PIC16F877 posee cuatro bancos de memoria RAM, cada banco posee 128 bytes. De estos 128 los primeros 32 (hasta el 1Fh) son registros que cumplen un propósito especial en el control del microcontrolador y en su configuración. Los 96 siguientes son registros de uso general que se pueden usar para guardar los datos temporales de la tarea que se está ejecutando, figura 2.5 Todas las posiciones o registros de memoria se pueden acceder directa o indirectamente (esta última forma a través del registro selector FSR). Para seleccionar que página o banco de memoria se trabaja en un momento determinado se utilizan los bits RP0 y RP1 del registro STATUS.

INDF	00h	INDF	80h	INDF	100h	INDF	180h
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
FCL	02h	PCL	82h	FCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR TRISA	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h		105h		185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
PORTC	07h	TRISC	87h		107h		187h
PORTD	08h	TRISD	88h		108h		188h
PORTE	09h	TRISE	89h		109h		189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	EEDATA	10Ch	EECON1	18Ch
PIR2	0Dh	PIE2	8Dh	EEADR	10Dh	EECON2	18Dh
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh	Reservado	18Eh
TMR1H	0Fh		8Fh	EEADRH	10Fh	Reservado	18Fh
T1CON	10h		90h		110h		190h
TMR2	11h	SSPCON2	91h	Registros de Propósito General 16 Bytes		Registros de Propósito General 16 Bytes	
T2CON	12h	PR2	92h				
SSPBUF	13h	SSPADD	93h				
SSPCON	14h	SSPSTAT	94h				
CCPR1L	15h		95h				
CCPR1H	16h		96h				
CCP1CON	17h		97h				
RCSTA	18h	TXSTA	98h				
TXREG	19h	SPBRG	99h				
RCREG	1Ah		9Ah				
CCPS2L	1Bh		9Bh	Registros de Propósito General 80 Bytes	11Fh 120h	Registros de Propósito General 80 Bytes	11Fh 1A0h
CCPS2H	1Ch		9Ch				
CCP2CON	1Dh		9Dh				
ADRESH	1Eh	ADRESL	9Eh				
ADCON0	1Fh	ADCON1	9Fh				
Registros de Propósito General 96 Bytes	20h		A0h				
		Registros de Propósito General 80 Bytes					
	7Fh		0EFh 0F0h				
Banco 0		Banco 1	FFh	Banco 2	17Fh	Banco 3	1FFh

FIG. 2.5 ORGANIZACIÓN DE LA MEMORIA RAM DEL PIC16F877

2.2.1.7.1 RESUMEN DE ALGUNOS DE LOS REGISTROS DE CONFIGURACIÓN

BANCO 0:

TMR0: Registro del temporizador/contador de 8 bits.

PCL: Byte menos significativo del contador de programa (PC).

STATUS: Contiene banderas (bits) que indican el estado del procesador después de una operación aritmética/lógica.

FSR: Registro de direccionamiento indirecto.

PORTA, PORTB, PORTC, PORTD, PORTE: Registro de puertos de E/S de datos. Conectan con los pines físicos del micro.

PCLATH: Byte alto (más significativo) del contador de programa (PC).

INTCON: Registro de control de las interrupciones.

ADRESH: Parte alta del resultado de la conversión A/D.

ADCON0: Controla la operación del módulo de conversión A/D

BANCO 1:

OPTION: Registro de control de frecuencia del TMR0.

TRISA, TRISB, TRISC, TRISD, TRISE: Registros de configuración de la operación de los pines de los puertos.

ADRESL: Parte baja del resultado de la conversión A/D.

ADCON1: Controla la configuración de los pines de entrada analoga.

BANCO 2:

TMR0: Registro del temporizador/contador de 8 bits.

PCL: Byte menos significativo del contador de programa (PC).

FSR: Registro de direccionamiento indirecto.

EEDATA: Registro de datos de la memoria EEPROM.

EEADR: Registro de dirección de la memoria EEPROM.

PCLATH: Byte alto (más significativo) del contador de programa (PC).

INTCON: Registro de control de las interrupciones.

BANCO 3:

OPTION: Registro de control de frecuencia del TMR0.

EECON1: Control de lectura/escritura de la memoria EEPROM de datos.

EECON2: No es un registro físico.

2.2.2 TECNOLOGÍA GSM

El sistema GSM (Global System for Mobile Communication) puede ser definido como un sistema de radio comunicaciones, porque está destinado para comunicaciones de media y larga distancia, digital, móvil y celular, lo interesante del sistema es que nos proporciona servicios de datos, e incluye integración de servicios.

“Este sistema nos proporciona una calidad de audio que ayuda a la fiabilidad de la comunicación, incluye sistemas de seguridad en el aspecto de autenticación de usuarios, actualmente este sistema es utilizado a nivel mundial por la mayoría de operadores celulares.”[5]

El roaming internacional es una de las armas poderosas con la que cuenta este sistema, especialmente utilizado por los usuarios de negocios que constantemente viajan al exterior, solamente con tener un móvil que trabaje en las cuatro bandas y un convenio internacional de operadoras el usuario accede de manera automática a este servicio.

El sistema GSM nos proporciona los siguientes servicios:

- El envío de voz
- la posibilidad de realizar una llamada de emergencia a un número común (911) en forma gratuita y con cualquier compañía de que tenga cobertura.
- el servicio de mensajes cortos (SMS) enviados por canales independientes de señalización,
- Envío y recepción de fax digital.

Además de estos servicios existen los servicios portadores, dentro de los que se pueden incluir la transmisión de datos a 9600 bits por segundo (bps), aunque con el desarrollo que han sufrido los terminales en los últimos años esta tasa de velocidad puede alcanzar valores superiores a los 64Kbps mediante la agrupación de canales.

2.2.2.1 ARQUITECTURA

La arquitectura del sistema GSM está organizada en subsistemas, componentes e interfaces, el sistema tiene tres subsistemas: la estación móvil (Móvil Station), el subsistema de estación base (BSS-Base Station Subsystem) y el subsistema de conmutación (NSS-Network and Switching Subsystem) todos estos subsistemas intercambian datos mediante interfaces.

La estación móvil se comunica con el subsistema de estación base mediante el interfaz llamado radio y este con el subsistema de conmutación mediante el denominado interfaz. Desde este último sistema se intercambian los datos con la red telefónica fija.

La estación suele presentar el único elemento del sistema que el usuario llega a ver. Además de las funciones básicas de radio y de proceso necesarias para acceder a la red a través de la interfaz de radio, una estación móvil debe proporcionar un interfaz al usuario, mediante dispositivos de envío y recepción de datos.

[5] G. COLOURIS, J. DOLLIMORE, TIM, KINDBERG, SISTEMAS DISTRIBUIDOS, Conceptos y Diseño, Tercera edición, Editorial Addison Wesley 2001, pág. 79

La BSS está en contacto con las centrales de conmutación del NSS, la función del subsistema de estación base es la conexión entre las estaciones móviles y el subsistema de conmutación, resumiendo como la conexión entre usuarios de telecomunicaciones.

La interfaz de radio del sistema GSM es una combinación del Acceso Múltiple por División en Frecuencia (FDMA) y Acceso Múltiple por División de Tiempo (TDMA), la comunicación la realiza mediante la división del ancho de banda de cada usuario en distintos canales lógicos, que se utilizan para el transporte de información entre usuarios o la información de control del mismo sistema.

Según el tipo de información transportada se definen dos tipos de canales lógicos: el canal de control y los canales de tráfico, los canales de tráfico se utilizan para transportar la información entre usuarios, mientras que la función de los canales de control es transferir la información de la señalización, estos canales de control se pueden dividir en canales de control comunes y canales de control dedicados.

Los canales de control comunes llevan la información relativa a todos los móviles que se encuentren en la celda, convirtiéndose en un vínculo importante para el transporte de datos a los usuarios, por ejemplo de la existencia de una llamada, de la asignación de un canal para que otro no lo solicite, en la siguiente tabla encontramos los canales comunes.

CANALES DE CONTROL COMUN	UTILIZADOS POR LA ESTACION BASE
NOMBRE	CONOCIDO COMO
CANAL DE CONTROL DIFUSIVO	BCCH-BROADCAST CONTROL CHANEL
CANAL DE BÚSQUEDA	PCH-PAGING CHANEL
CANAL DE ACCESO ALEATORIO	RACH-RANDOM ACCESS CHANNEL
CANAL DE ACCESO GARANTIZADO	AGCH-ACCESS GRANT CHANNEL

TABLA 2.3 CANALES DE CONTROL COMÚN UTILIZADOS POR LA ESTACIÓN BASE

Los canales de control dedicados a un móvil llevan la información referente y orientada para un único Terminal, en la siguiente tabla especificamos los canales dedicados.

CANALES DE CONTROL	UTILIZADOS POR LA ESTACION BASE
NOMBRE	CONOCIDO COMO
CANAL DE CONTROL DEDICADO	SDCCH-STAND-ALONE DEDICATED
CANAL DE CONTROL ASOCIADO	SACCH-SLOW ASSOCIATED CONTROL
CANAL DE CONTROL ASOCIADO	FACCH-FAST ASSOCIATED CONTROL

TABLA 2.4 CANALES DE CONTROL DEDICADOS UTILIZADOS POR LA ESTACIÓN BASE

El Subsistema de Conmutación (NSS) incluye las funciones básicas de conmutación del GSM, así como las bases de datos para los usuarios y la gestión de movilidad, siendo su función principal el de gestionar las comunicaciones entre los usuarios GSM y los usuarios de otras redes de telecomunicación, como puede ser para otras redes de sistemas móviles en la Figura 2 se detalla la arquitectura del sistema GSM.

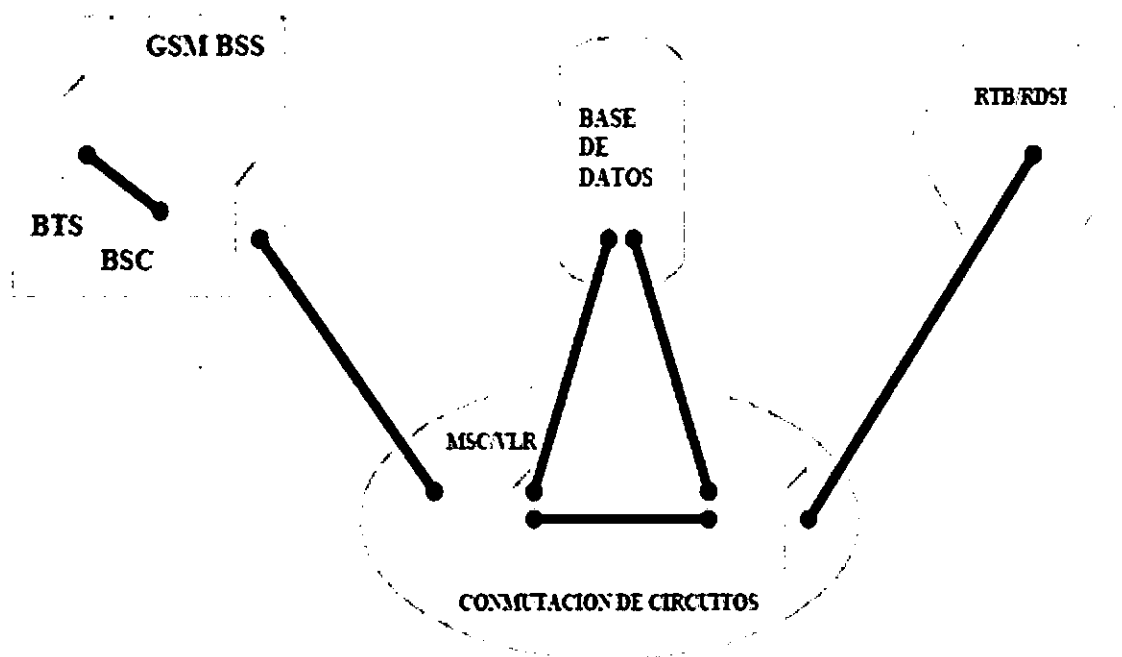


FIG. 2.6 ARQUITECTURA DEL SISTEMA GSM

Con los sistemas analógicos cualquier persona con una estación de radioaficionado o con un escáner de radio escuchaba las conversaciones del vecino, pero con el sistema GSM esta tarea se vuelve complicada y queda reservada a equipos sofisticados ya que en la conversación digital el escáner tiene que ser conectado a un ordenador que tendrá el trabajo de interpretar una señal que está digitalizada, entrelazada y comprimida esta información por lo general está cifrada mediante claves y ciertos algoritmos de encriptación que en principio se creían indescifrables. En el sistema GSM existe un centro de autenticación que es el encargado de autorizar al móvil a introducirse a la red, en donde cada usuario tiene una clave secreta la cual es asignada en el momento de contratar el servicio y está localizada en dos sitios en la SIM card y en el móvil siendo el sistema el encargado de verificar las claves personales y de realizar la respectiva autenticación.

2.2.2.2 DESARROLLO DE UNA LLAMADA GSM

En la siguiente grafica se detalla un ejemplo de enrutamiento de llamadas para GSM.

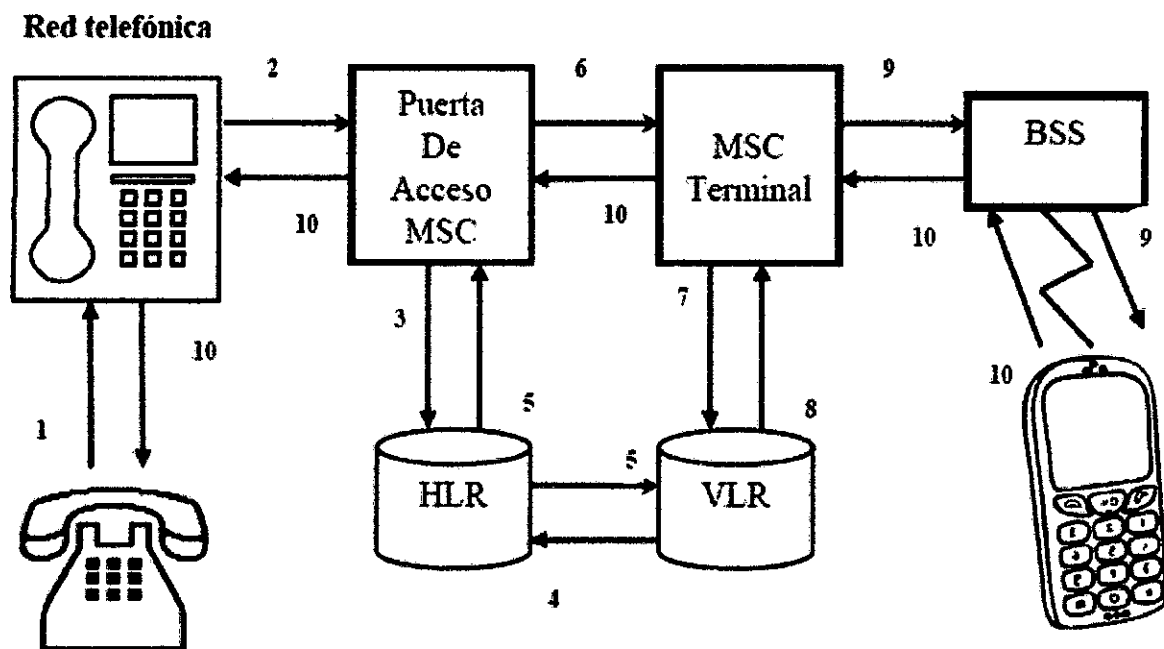


FIG. 2.7 EJEMPLO DE UN ENRUTAMIENTO DE LLAMADAS GSM

- Paso 1: Un usuario de telefonía llama a la unidad móvil a través de la red pública.
- Paso 2: La llamada se enruta a un MSC, el que se encarga de examinar los dígitos marcados y determina que no puede enrutar la llamadas más lejos.
- Paso 3: El registro de ubicación HLR interroga al usuario llamado.
- Paso 4: El HLR interroga el registro de ubicación del visitante VLR que actualmente está dando servicio al usuario.
- Paso 5: El VLR devuelve un número de enrutamiento al HLR que lo devuelve al MSC.
- Paso 6: Con este número de enrutamiento, el MSC enruta la llamada al MSC Terminal.
- Paso 7: El MSC pide al VLR correlacionar la llamada con el subscritor.
- Paso 8: El VLR realiza la acción que se le solicite.
- Paso 9: La BSS recibe una solicitud de notificación del MSC Terminal y envía una señal de notificación.
- Paso 10: Cuando la señal de usuario regresa, la llamada se completa.

2.2.2.3 VENTAJAS Y DESVENTAJAS ANTE OTRAS TECNOLOGÍAS

2.2.2.3.1 VENTAJAS

- GSM a nivel mundial tiene un mayor número de usuarios que CDMA.
- GSM cuenta con el mayor número de operadoras celulares a nivel mundial.
- GSM incorpora en su sistema GPRS una ventaja tecnológica para brindar servicios 3G.
- Con la implementación de la banda GSM de 850 Mhz permitió el crecimiento de esta tecnología en Latinoamérica.
- Roaming internacional.
- Tecnología de implementación relativamente económica en relación a CDMA lo cual permite obtener rentabilidad económica en poco tiempo.
- Uso eficiente del espectro radioeléctrico.
- Facilidad para la transmisión de datos inalámbricos.
- GSM brinda claridad de voz en las llamadas.
- GSM da la facilidad de cambiar de dispositivo móvil mediante el SIM.
- GSM puede operar en cuatro bandas 850 Mhz, 900 Mhz, 1800 Mhz, 1900 Mhz.
- GSM permite el envío y recepción de información multimedia.
- GSM tiene su arquitectura abierta lo cual brinda una compatibilidad con otras tecnologías.

2.2.2.3.2 DESVENTAJAS

- GSM a nivel tecnológico es inferior a CDMA.
- CDMA utiliza el espectro radioeléctrico de una manera más eficiente que GSM.
- GSM no posee el nivel de seguridad que tiene CDMA.
- En CDMA las comunicaciones son codificadas con lo cual se logra una mayor cantidad de enlaces.
- CDMA posee un sistema que le permite una mayor velocidad en la transmisión de datos más o menos unos 144 Kbps.
- CDMA no es propenso a interferencia externa.
- GSM requiere un número considerado de radio bases para brindar una buena cobertura.

2.2.2.4 LIMITACIONES DE GSM PARA LA TRANSMISIÓN DE DATOS

Las redes GSM tienen ciertas limitaciones para la transmisión de datos:

- Velocidad de transferencia de 9,6 Kbps.
- Tiempo de establecimiento de conexión, de 15 a 30 segundos.
- Además las aplicaciones deben ser reinicializadas en cada sesión.
- Pago por tiempo de conexión.
- Problemas para mantener la conectividad en itinerancia (Roaming).

2.2.3 TECNOLOGÍA GPRS

2.2.3.1 ¿POR QUÉ SURGE GPRS?

Hoy en día el número de usuarios de telefonía móvil y de usuarios de Internet ha crecido de una manera increíble. Debido a esto era inevitable que en algún momento ambos mundos se fusionasen. Al sistema GPRS se le conoce también como GSM-IP ya que usa la tecnología IP (Internet Protocol) para acceder directamente a los proveedores de contenidos de Internet.

2.2.3.2 ¿QUÉ ES GPRS?

GPRS es una nueva tecnología que comparte el rango de frecuencias de la red GSM utilizando una transmisión de datos por medio de 'paquetes'.

“La conmutación de paquetes es un procedimiento más adecuado para transmitir datos, hasta ahora los datos se habían transmitido mediante conmutación de circuitos, procedimiento más adecuado para la transmisión de voz.”[6]

Los canales se comparten entre los diferentes usuarios

En GSM, cuando se realiza una llamada se asigna un canal de comunicación al usuario, que permanecerá asignado aunque no se envíen datos. En GPRS los canales de comunicación se comparten entre los distintos usuarios dinámicamente, de modo que un usuario sólo tiene asignado un canal cuando se está realmente transmitiendo datos. La tecnología GPRS, o generación 2.5, representa un paso más hacia los sistemas inalámbricos de Tercera Generación o UMTS. Su principal baza radica en la posibilidad de disponer de un terminal permanentemente conectado, tarificando únicamente por el volumen de datos transferidos (enviados y recibidos) y no por el tiempo de conexión como hemos podido observar en un punto anterior.

Obtiene mayor velocidad y mejor eficiencia de la red.

Tradicionalmente la transmisión de datos inalámbrica se ha venido realizando utilizando un canal dedicado GSM a una velocidad máxima de 9.6 Kbps. Con el GPRS no sólo la velocidad de transmisión de datos se ve aumentada hasta un mínimo 40 Kbps y un máximo de 115 Kbps por comunicación, sino que además la tecnología utilizada permite compartir cada canal por varios usuarios, mejorando así la eficiencia en la utilización de los recursos de red. La tecnología GPRS permite proporcionar servicios de transmisión de datos de una forma más eficiente a como se venía haciendo hasta el momento.

2.2.3.3 ¿POR QUÉ ES MEJOR GPRS QUE GSM?

Como hemos visto anteriormente el sistema GSM no se adaptaba del todo bien a la transmisión de datos. Vamos a ver ahora las características de GPRS:

- Velocidad de transferencia de hasta 144 Kbps.

[6] Ferran Casadevall y Oriol Sallent. "Sistema GPRS". UPC Barcelona. Mayo 2002. Pág. 17

- Conexión permanente. Tiempo de establecimiento de conexión inferior al segundo.
- Pago por cantidad de información transmitida, no por tiempo de conexión. Veamos unos ejemplos de los tamaños de información que descargaríamos:
- Envío de un e-mail de 5 líneas de texto con un anexo (documento tipo Word de 4 páginas), consumiría alrededor de 95 kbytes.
- Acceder a un buscador, buscar un término (ej. viajes) y recibir una pantalla de respuesta podría ocupar 100 kbytes aproximadamente.
- Recibir una hoja de cálculo (documento tipo Excel de 5 hojas), consumiría aproximadamente 250 kbytes.
- Bajarse una presentación (documento tipo PowerPoint de 20 diapositivas y con fotos) equivale a unos 1.000 kbytes.

Como vemos estas características se amoldan mucho mejor para la transmisión de datos que el tradicional sistema GSM.

2.2.3.4 VENTAJAS DEL GPRS PARA EL USUARIO

Las ventajas que obtiene el usuario con el sistema GPRS son consecuencia directa de las características vistas en el punto anterior. Característica de "Siempre conectado"; un usuario GPRS puede estar conectado todo el tiempo que desee, puesto que no hace uso de recursos de red (y por tanto no paga) mientras no esté recibiendo ni transmitiendo datos.

“Tarificación por volumen de datos transferidos, en lugar de por tiempo. Coste nulo de establecimiento de conexión a la red GPRS, frente a los quantum de conexiones existentes actualmente en GSM. “[7]

Mayor velocidad de transmisión. En GSM sólo se puede tener un canal asignado (un "timeslot"), sin embargo, en GPRS, se pueden tener varios canales asignados, tanto en el sentido de transmisión del móvil a la estación base como de la estación base al móvil. La velocidad de transmisión aumentará con el número de canales asignados. Además, GPRS permite el uso de esquemas de codificación de datos que permiten una velocidad de transferencia de datos mayor que en GSM.

Posibilidad de realizar/recibir llamadas de voz mientras se está conectado o utilizando cualquiera de los servicios disponibles con esta tecnología. Modo de transmisión asimétrico, más adaptado al tipo de tráfico de navegación HTML o WML (un terminal GPRS 4+1 (4 slots downlink y 1 uplink) tendrá cuatro veces mayor capacidad de transmisión de bajada que de subida).

[7] Ferran Casadevall y Oriol Sallent. "Sistema GPRS". UPC. Barcelona. Mayo 2002. Pág. 21

2.2.3.5 SERVICIOS DEL GPRS PARA EL USUARIO

Los servicios que obtendrá un usuario de este sistema serían los equivalentes a tener un PC conectado a Internet, siendo este de tamaño bolsillo.

2.2.3.5.1 ACCEDER EN MOVILIDAD A INTERNET Y CORREO ELECTRÓNICO

GPRS permite acceder en movilidad a todas las facilidades de Internet usando el terminal GPRS como módem:

- Acceso a cuentas de correo Internet (lectura y envío de e-mails).
- Aviso de recepción de correo en el móvil.
- Navegación por Internet.
- Descarga de ficheros.
- Desde cualquier PC, asistente personal digital (PDA) o directamente desde el terminal GPRS (si sus características lo permiten), pagando sólo por el volumen de datos transmitidos y recibidos y no por el tiempo de conexión.

2.2.3.5.2 ACCESO A CUENTAS DE CORREO CORPORATIVAS (INTRANET)

- GPRS permite utilizar desde un dispositivo móvil (Ordenador portátil, PDA o el propio móvil) los sistemas de correo electrónico de la empresa (Microsoft Mail, Outlook Express, Microsoft Exchange, Lotus Notes etc...).
- El usuario puede acceder en movilidad a su correo corporativo, leerlo y contestarlo como si estuviera en la oficina.

2.2.3.5.3 ACCESO A BASES DE DATOS Y APLICACIONES CORPORATIVAS DESDE UN DISPOSITIVO MÓVIL

- Gestión de Fuerza de Ventas: consulta de estados de pedidos, consulta de catálogos, consulta de stocks, información relativa a los clientes... desde cualquier lugar.
- Gestión de equipos de trabajo que operan fuera de la empresa (equipos de mantenimiento, supervisión, reparto). Con GPRS se pueden enviar avisos, cumplimentar partes de trabajo, obtener información detallada sobre envíos o reparaciones desde cualquier lugar.

2.2.3.5.4 ACCESO GPRS A APLICACIONES WAP PARA USO EMPRESARIALES (A TRAVÉS DEL SERVICIO WAP)

- Agenda
- Directorios
- Tarjetas de visita
- E-mail, correo
- Tareas
- Tablón
- Enviar fax

- Gestión de equipos.

2.2.3.5.5 ACCESO A SERVICIOS DE INFORMACIÓN (A TRAVÉS DEL SERVICIO WAP)

- Canales temáticos: Noticias, Finanzas, Viaje.
- Guía Conecta: Guía de carreteras, Reserva de restaurantes, Guía de teléfono.
- Centro comercial: Banca móvil, Entradas. Internet/ Servicios: Buscador, Traductor.

2.2.3.6 VENTAJAS DEL GPRS PARA LA OPERADORA

Uso eficiente de los recursos de la red: los usuarios sólo ocupan los recursos de la red en el momento en que están transmitiendo o recibiendo datos, y además se pueden compartir los canales de comunicación entre distintos usuarios y no dedicados como en el modelo GSM.

2.2.3.7 ¿CÓMO SE ACCEDE A GPRS?

Ya existen en el mercado un buen número de móviles adaptados al sistema GPRS. Los terminales GPRS presentan las siguientes características comunes:

Capacidad Dual:

Los terminales GPRS están adaptados para aprovechar la cobertura existente GSM para la voz y en GPRS para la transmisión de datos.

Velocidad de transferencia:

Los terminales GPRS utilizan varios canales simultáneos o slots.

El número de canales depende de cada terminal, variando de 1 a 4 para la recepción de datos y de 1 a 2 para el envío.

Cada canal representa una velocidad teórica de 13.4 kilobits (en GSM sólo 9 Kbits).

Tarjeta SIM:

La tarjeta SIM es la misma que para GSM. No es preciso cambiar de tarjeta para usar GPRS. Existen tres tipos de terminales, cada uno con sus características.

2.2.3.8 ¿CÓMO FUNCIONA GPRS?

PILA DE PROTOCOLOS DEL PLANO DE TRANSMISIÓN

El plano de transmisión es el encargado de proveer la transmisión de los datos del usuario y su señalización para el control de flujo, detección de errores y la corrección de los mismos.

GTP: GPRS TUNNELING PROTOCOL. Es el encargado de transportar los paquetes del usuario y sus señales relacionadas entre los nodos de soporte de GPRS (GSN). Los paquetes GTP contienen los paquetes IP o X.25 del usuario. Por debajo de él, los protocolos estándares TCP o UDP se encargan de transportar los paquetes por la red. Resumiendo, en el Backbone del GPRS tenemos una arquitectura de transporte IP/X.25-sobre-GTP-sobre-UDP/TCP-sobre IP.

SNDCP (Subnetwork Dependent Convergence Protocol). Es el encargado de transferir los paquetes de datos entre los SGSN (nodo responsable de la entrega de paquetes al terminal móvil) y la estación móvil. Las funciones que desempeña:

- Multiplexación de diversas conexiones de la capa de red en una conexión lógica virtual de la capa LLC.
- Compresión y descompresión de los datos e información redundante de cabecera.

AIR INTERFACE: Conciernen a las comunicaciones entre la estación móvil y la BSS en los protocolos de las capas física, MAC, y RLC.

Las subcapas RLC/MAC permiten una eficiente multiplexación multiusuario en los canales de paquetes de datos compartidos, y utiliza un protocolo ARQ selectivo para transmisiones seguras a través del interfaz aire. El canal físico dedicado para tráfico en modo paquete se llama PDCH (Packet Data Channel).

En adelante se considerará la capa de enlace de datos (Data Link Layer) y la capa física (Physical Layer) como parte del Interfaz Aire Um.

DATA LINK LAYER: Capa de enlace de datos. Se encuentra entre la estación móvil (el móvil GPRS en sí) y la red.

Se subdivide en:

- La capa LLC (entre MS-SGSN): Provee un enlace altamente fiable y está basado en el protocolo DIC e incluye control de secuencia, entrega en orden, control de flujo, detección de errores de transmisión y retransmisión. Es básicamente una adaptación del protocolo LAPDm usado en GSM.
- La capa RLC/MAC (entre MS-BSS): Incluye dos funciones. El principal propósito de la capa de Control de Radio Enlace (RLC) es la de establecer un enlace fiable. Esto incluye la segmentación y reensamblado de las tramas LLC en bloques de datos RLC y ARQ (peticiones de retransmisión) de códigos incorregibles. La capa MAC controla los intentos de acceder de un MS a un canal de radio compartido por varios MS.

PHYSICAL LAYER: Capa física entre MS y BSS. También se subdivide en dos subcapas.

- La capa del enlace físico (PLL) provee un canal físico. Sus tareas incluyen la codificación del canal (detección de errores de transmisión, corrección adelantada (FEC), indicación de códigos incorregibles), interleaving y la detección de congestión del enlace físico.
- La capa de enlace de radio frecuencia (RFL) trabaja por debajo de la PLL e incluye la modulación y la demodulación.

INTERFAZ BSS-SGSN: El protocolo de aplicación BSS GPRS (BSSGP) se encarga del enrutado y lo relativo a la información de la QoS entre BSS y SGSN. El servicio de red (NS) está basado en el protocolo de Frame Relay.

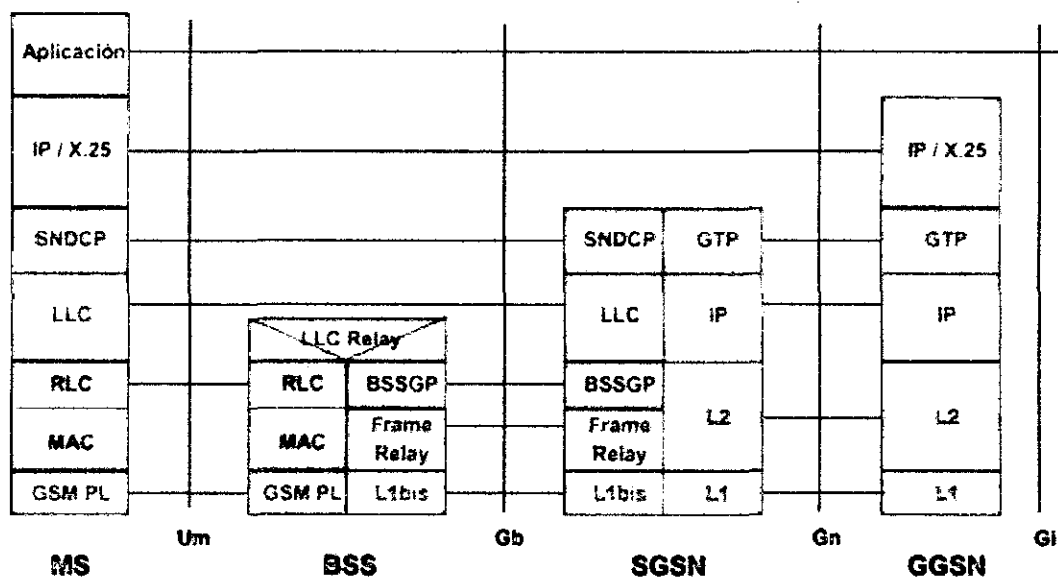


FIG. 2.8 PILA DE PROTOCOLOS DEL PLANO DE SEÑALIZACIÓN

Se incluye en esta pila de protocolos aquellos encargados del control y mantenimiento de las funciones del plano de transmisión, conexión desconexión, activación de contexto, control de caminos de routing y localización de los recursos de la red.

GMM/SM: GPRS MOBILITY MANAGEMENT/SESSION MANAGEMENT. Es el protocolo que se encarga de la movilidad y la gestión de la sesión en momentos de la ejecución de funciones de seguridad, actualizaciones de rutas, etc.

La señalización entre SGSN y los registros HLR, VLR, y EIR utilizan los mismos protocolos que GSM con ciertas funciones ampliadas para el funcionamiento con el GPRS

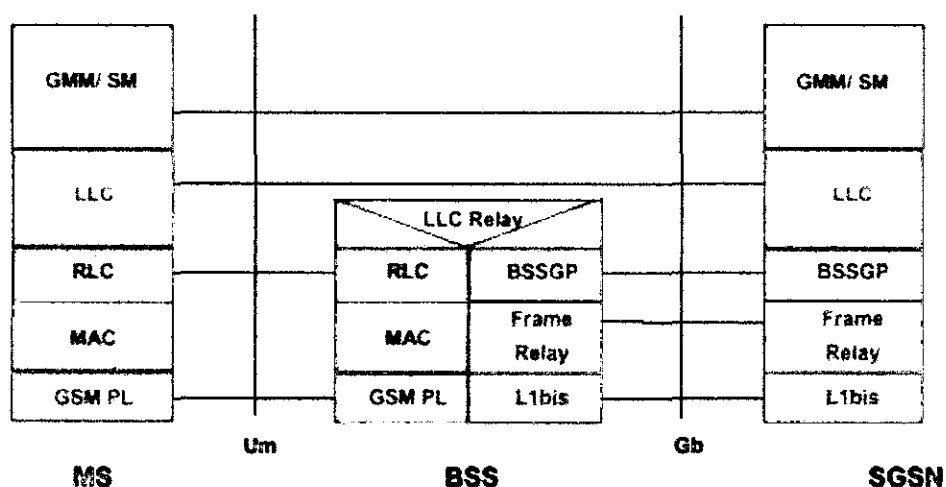


FIG. 2.9 PROTOCOLOS GPRS

2.2.3.9 MÓDULO GPRS/GSM (CELULAR) SIM900

El módulo GSM/GPRS se basa en el controlador SIM900 de SIMCOM y está diseñado para trabajar con Arduino y sus versiones compatibles. Este módulo permite comunicarte usando la red GSM de telefonía celular. Con este podrás acceder a los servicios SMS, MMS, GPRS y Telefonía de una manera sencilla enviando comandos AT. Asimismo tiene incorporadas en la placa 12 GPIOs, 2 PWM y un ADC propios del módulo SIM900.

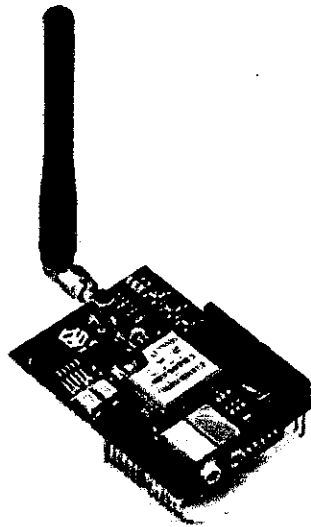


FIG. 2.10 MÓDULO GPRS/GSM SIM 900

2.2.3.10 ESPECIFICACIONES TÉCNICAS

Quad-Band 850 / 900/ 1800 / 1900 MHz - funciona en todas las redes celulares del planeta.

GPRS multi-slot class 10/8

Estación móvil GPRS clase B

Cumple con GSM phase 2/2+

Clase 4 (2 W @ 850 / 900 MHz)

Clase 1 (1 W @ 1800 / 1900MHz)

Controlable vía comandos AT estándar: GSM 07.07 & 07.05 Comandos mejorados:

SIMCOM AT Commands.

Servicio corto de mensajes (SMS): para poder enviar pequeños paquetes de datos a través de la red celular.

Pila TCP/UDP incorporada: permite enviar datos a un servidor web.

Incorpora un reloj en tiempo real: RTC (Requiere pila)

Puerto serial configurable para comunicación con el módulo.

2.2.3.11 DIAGRAMA FISICO DEL HARDWARE Y DESCRIPCIÓN

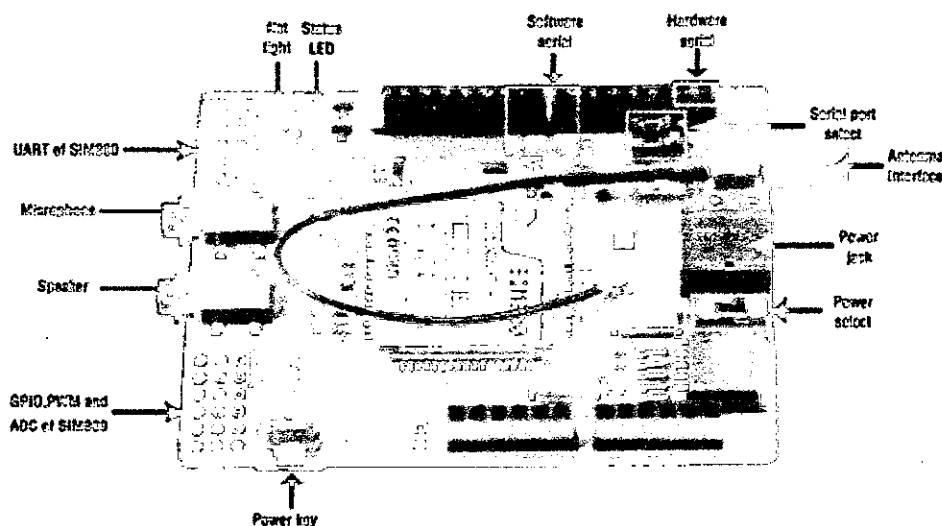


FIG. 2.11 DIAGRAMA FISICO DEL MODULO SIM 900

- Selección de alimentación: Selecciona el suministro eléctrico del módulo GPRS (La alimentación es de 5V)
- Jack de alimentación: Conexión externa de 4.8V a 5V DC de alimentación
- Interface de antena: Conexión con antena externa
- Selección de puerto serial: Selección de puerto serie a través de software y hardware que están conectados con el módulo GPRS
- Hardware Serial: D0/D1 de Arduino
- Software serial: D7/D8 de Arduino
- Estado de LED: Indica el encendido del modulo
- Net light: Indica la vinculación del módulo SIM900 con la red
- UART of SIM900 - UART pins breakout of SIM900
- Micrófono: Para responder la llamada telefónica
- Speaker: Para escuchar la llamada telefónica
- GPIO, PWM and ADC of SIM900 - GPIO, PWM and ADC pins breakout of SIM900
- Power key - power up and down for SIM900
- D0: D1: Sin usar se puede seleccionar el puerto para comunicación serie a través de software con el modulo
- D2: D3: D4: D5: D6: Sin usar
- D7: D8: Usar si selecciona el puerto serie para comunicación con el módulo
- D9: Usado para control por software the power up or down of the SIM900
- D10: D11: D12: D13: Sin usar
- D14 (A0): D15(A1): D16(A2): D17(A3): D18(A4): D19(A5): Sin usar

2.2.3.12 DIAGRAMA DE BLOQUES DEL MODULO SIM 900

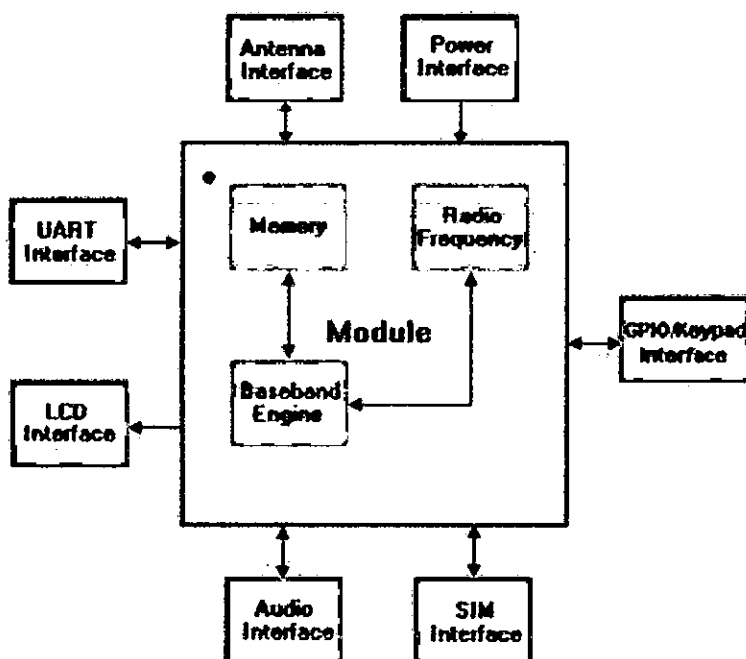


FIG. 2.12 DIAGRAMA LOGICO DEL SIM 900

La siguiente figura muestra un diagrama lógico del SIM900 e ilustra principalmente la parte funcional:

- El motor de banda base GSM
- El destello y SRAM
- La parte de radiofrecuencia GSM
- La interfaz de la antena
- Las otras interfaces

2.2.3.13 EVALUACIÓN DEL SIM900

Para ayudarle en la aplicación de SIM900, SIMCOM puede suministrar una Junta Directiva de Evaluación (EVB) que interconecta el SIM900 directamente con suministro de fuerza apropiado, agarradera de la tarjeta SIM, puerto RS232 serial portuario, del microteléfono del puerto, del auricular, línea en puerto, antena y todo GPIO del SIM900.

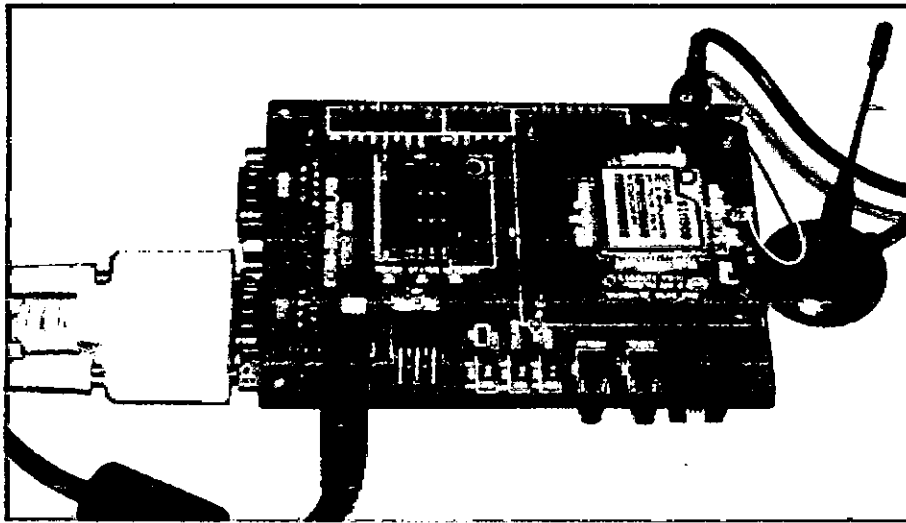


FIG. 2.13 FOTOGRAFIA DEL MODULO SIM 900

2.2.4 BLUETOOTH

El bluetooth, propiamente dicho, es un sistema de transmisión de datos entre dispositivos a corta distancia que no necesita de cables, ya que la transferencia se hace de manera inalámbrica. La banda de transmisión de datos en la que se mueve es la que va entre los 2,4 a 2,48 GHz de amplio espectro, con posibilidad de transmitir hasta 1600 saltos/s. con un total de 79 frecuencias con intervalos de 1Mhz.



FIG. 2.14 SIMBOLO DE BLUETOOTH

Esto es lo que viene a ser la definición de bluetooth de una manera técnica, podríamos decir que:

“Bluetooth es un sistema de enviar y recibir archivos, música, datos, vídeo, imágenes o incluso voz sin necesidad de cables y a una corta distancia”. [8]

Además a través de la conexión bluetooth, podremos conectarnos de manera inalámbrica a otro dispositivo para compartir la conexión de Internet.

2.2.4.1 HISTORIA DE BLUETOOTH

El nombre de bluetooth tiene un origen de procedencia nórdica, concretamente del rey de Noruega y Dinamarca llamado Harald Blåtand, cuya traducción literal al inglés sería la de Harold Bluetooth. Este noble personaje fue conocido por ser un buenísimo comunicador, el cual hizo la gran gesta de unir las diferentes tribus noruegas, suecas y danesas.

2.2.4.2 PRINCIPIOS DEL BLUETOOTH

La primera empresa que creó un equipo de trabajo para investigar sobre sistemas de comunicación entre dispositivos, fue Ericsson, que allá por el año 1994 comenzó la investigación de una nueva interfaz de bajo consumo y coste, destinada al envío y recepción de datos entre teléfonos móviles y otros dispositivos.

No fue hasta el año 1999, cuando se creó el SIG de Bluetooth (Special Interest Group), el cual consistía en la unión de diferentes empresas, entre las cuales se encontraban en un primer momento Ericsson, Intel, Nokia, Toshiba e IBM. A estas empresas fundadoras del grupo, tan solo unos meses después se les unieron otras empresas de la tecnología tan importantes como Microsoft, 3COM, Motorola y Lucent.

2.2.4.3 VERSIONES DE BLUETOOTH

BLUETOOTH V1.0 Y V1.0B

Los primeros emisores receptores de bluetooth, fueron el v1.0 y v1.0B, los cuales ya están prácticamente obsoletos, y dieron muchísimos problemas a los fabricantes de teléfonos para la interacción entre dispositivos de diferentes compañías, asimismo, tenían el gran defecto que en cada transmisión de datos se enviaba nuestra dirección privada de dispositivo bluetooth, perdiendo así el anonimato que nos pudiese brindar este tipo de conexión inalámbrica.

BLUETOOTH V1.1

- Usa el estándar IEEE 802.15.1-2002
- Corregidos errores de las versiones anteriores.
- Canales no encriptados añadidos y soportados.
- Añadido el indicador de la señal o también denominado (RSSI)

[8] [http:// www.lcd.usb.ve/figueira/cursos/redes2/EXPO-em01/Bluetooth/bluetooth.htm](http://www.lcd.usb.ve/figueira/cursos/redes2/EXPO-em01/Bluetooth/bluetooth.htm)

BLUETOOTH V1.2

- Compatibilidad con usb 1.1.
- Mejora la velocidad de conexión y transferencia de datos.
- Añadida la función de detección de otros dispositivos bluetooth en el radio de actuación.
- Notables mejoras en la calidad del audio.
- Host Controller Interface (HCI)
- Nuevo protocolo estándar IEEE 802.15.1-2005.
- Añadido control de flujo y modos de retransmisión L2CAP.

BLUETOOTH V2.0 + EDR

La mejora implementada en esta nueva versión, hace referencia a la opción del propio fabricante del dispositivo de incorporar la EDR (Enhanced Data Rate), esto no viene a significar que todos los dispositivos 2.0 vengán con este sistema de transmisión de datos a mayor velocidad, ya que como he comentado es de carácter opcional. Su transferencia máxima de datos es de 3Mb/s. aunque su tasa real máxima sea la de 2.1Mb/s. esta versión mantiene la compatibilidad con la versión anterior de la interfaz bluetooth.

BLUETOOTH V2.1 + EDR

Las mejoras de esta nueva versión, son mejoras sustanciales siempre mirando hacia la seguridad de nuestros datos, así de esta manera se ha añadido Secure Simple Pairing (SSP), lo que permite un mejor filtrado de nuestros datos y una seguridad superior a la de la versión anterior. A su vez, se ha mejorado notablemente el consumo de energía, gracias a la nueva tecnología oler subrating.

BLUETOOTH V3.0 + HS

Este nuevo modelo de la interfaz, fue lanzado en abril de 2009, y su mayor logro es el aumento de la velocidad de transmisión de datos hasta los 24Mb/s., además de incluir una nueva característica la cual hace uso del wifi para el envío y recepción de grandes paquetes de datos, usando el estándar 802.11 de alta velocidad, esta nueva característica es denominada Alternativa MAC / PHY

BLUETOOTH V4.0

Esta versión es la más reciente de todas, y fue lanzada en el año 2010, combina la tecnología bluetooth clásica con la conexión inalámbrica vía wifi, para dotar a los dispositivos en los que vienen instalados de una velocidad de emisión y transferencia de datos de nada más y nada menos que de 32Mb/s.. Esta nueva interfaz de bluetooth viene incluida en los más avanzados Smartphones y dispositivos tecnológicos de última generación.

BLUETOOTH V4.1

Los responsables del desarrollo de la especificación Bluetooth han presentado una nueva versión. La actualización 4.1 que incorpora novedades importantes de cara al usuario al facilitar

la reconexión entre sus dispositivos una vez que estos salen y vuelven a entrar en el radio de acción, y que claramente orienta su uso al internet de las cosas.

Estas novedades facilitan la conexión y reconexión de dispositivos, como hemos dicho, sin la necesidad de que el usuario deba hacer nada, al menos una menor frecuencia de interacción. Además permitirá que un dispositivo funcione tanto como periférico y hub de datos a la par, se mejora el soporte para convivir con otros protocolos como LTE, intercambio de datos más eficientes, etc.

BLUETOOTH V4.2

De acuerdo con un comunicado de prensa oficial, la versión 4.2 de la especificación básica Bluetooth salió cerca del final de 2014. Director Ejecutivo del Bluetooth Special Interest Group (SIG) Mark Powell dice que la actualización 4.2 espera continuar haciendo Bluetooth Smart "de la mejor solución para conectar toda la tecnología en su vida. "Bluetooth Core Especificación 4.2 esperanzas para dar a los desarrolladores y fabricantes más oportunidades de usar Bluetooth y construir una mejor experiencia de usuario para sus consumidores.

2.2.4.4 MÓDULO BLUETOOTH SLAVE (HC-06)

2.2.4.4.1 DESCRIPCIÓN GENERAL:

El módulo Bluetooth HC-06 es un dispositivo económico y sencillo de utilizar. Se puede utilizar para dar conectividad inalámbrica a través de una interfaz serial TTL entre Microcontroladores (PIC, Arduino) y otros dispositivos como PC, laptops o celulares Smartphone.

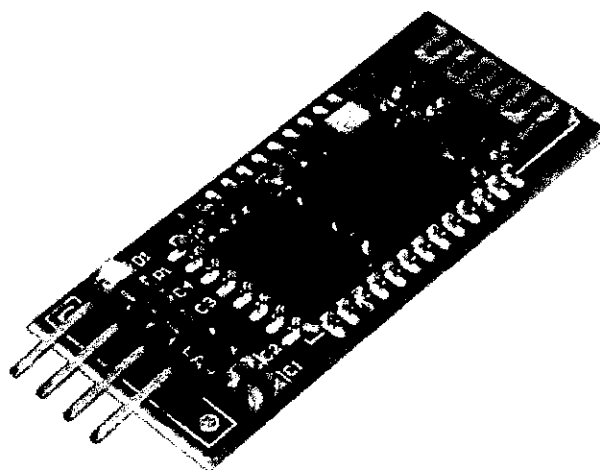


FIG. 2.15 MODULO BLUETOOTH

2.2.4.4.2 ESPECIFICACIONES:

- Protocolo: Bluetooth v2.0 + EDR
- Frecuencia: Banda ISM de 2,4 GHz
- Modulación: GFSK (Gaussian Frequency Shift Keying)
- Potencia de emisión: 4 dBm, clase 2
- Sensibilidad: -84dBm a 0.1% VER
- Velocidad asíncrona: 2.1Mbps (máx.) / 160 kbps.
- Velocidad síncrona: 1Mbps/1Mbps
- Alcance 10 mts
- Seguridad: Autenticación y encriptación
- Interfaz: Bluetooth - Puerto serie UART TTL
- Suministro de energía: 5VDC 50 mA
- Temperatura de trabajo: $-20 \sim +75^{\circ}\text{C}$
- Dimensión: 26.9mm x 13mm x 2,2 mm

2.2.5 MEDICIÓN DE LA EMISIÓN DE GASES EN VEHÍCULOS

2.2.5.1 DETECCIÓN DE GASES

La contaminación ambiental afecta el aire que respiramos, el agua que bebemos y los alimentos que consumimos. También afecta la producción de alimentos, la calidad general de nuestro ambiente circundante y puede poner en riesgo nuestra salud y bienestar. El control de la contaminación ambiental es necesario en casi todas las comunidades y países para proteger la salud de la población.

Diferentes tipos de gases nos rodean continuamente. La composición del aire en condiciones normales está formada por una mezcla de gases cuyas proporciones son las siguientes:

- Nitrógeno (N_2): 78,084%
- Oxígeno (O_2): 20,946%
- Argón (Ar): 0,934%
- Dióxido de carbono (CO_2): 0.0314%

Otros gases como el neón, metano, helio, hidrógeno, conforman el 0,0046% restante. Cualquier variación significativa en estos porcentajes, implica una posible acumulación de gas nocivo que puede llegar a ser perjudicial para las personas y medio ambiente.

Es por este motivo que existen una gran variedad de detectores de gases diseñados para evitar o advertir de situaciones de riesgo potencial, y que asociados al sistema electrónico correspondiente, permiten que este pueda dar respuestas luminosas, acústicas, de corte, ventilación y extracción.

Estos sistemas electrónicos de detección de gases pueden estar diseñados para medir un solo gas o para medir varios gases de forma simultánea, con el objetivo de minimizar los incidentes relacionados con la contaminación del medio ambiente, asfixia y explosiones.

2.2.5.2 MONÓXIDO DE CARBONO

El monóxido de carbono es un gas altamente tóxico, en condiciones normales (25°C y 760 mm Hg) es incoloro e inodoro, que no se puede ver ni oler, pero que puede causar la muerte cuando se lo respira en niveles elevados. Es muy estable y tiene una vida media en la atmósfera de 2 a 4 meses. En la actualidad las emisiones de CO son grandes. El CO se produce cuando se queman materiales combustibles como gas, gasolina, kerosene, carbón, petróleo o madera. Las chimeneas, las cocinas de carbón, los calentadores de agua, y los aparatos domésticos que queman combustible, como las hornillas de las cocinas, los autos detenidos con el motor encendido.

2.2.5.3 MEDICIÓN DE LAS CONCENTRACIONES DE UN GAS

2.2.5.3.1 PARTES POR MILLÓN

Partes por millón (ppm) es la unidad de medida con la que se evalúa la concentración. Se refiere a la cantidad de unidades de la sustancia que hay por cada millón de unidades del conjunto.

Ejemplo:

Una botella de gas contiene 250 ppm de monóxido de carbono (CO), esto significa que si se extraen de esta botella un millón de moléculas de gas, 250 de ellas serán moléculas de monóxido de carbono. El resto de las 99.750 moléculas de dióxido de nitrógeno y de oxígeno.

2.2.5.4 CLASIFICACIÓN DE ESTADOS DE LA CALIDAD DEL AIRE SEGÚN DIGESA, SENAMHI, OMS, OSHA A NIVEL NACIONAL

Las guías para la calidad del aire de la Organización Mundial de la Salud (OMS) y de la Administración de Seguridad y Salud Ocupacional (OSHA) proporcionan una base para proteger la salud pública de los efectos adversos contaminantes ambientales y para eliminar o minimizar los contaminantes conocidos o los que se consideran probablemente peligrosos para la salud y el bienestar humano. Las guías también ayudan a los gobiernos a implementar medidas de control de la contaminación del aire en el nivel local.

Los valores presentados por la OMS y la OSHA proporcional a DIGESA Y SENAMHI los niveles de contaminación del aire por debajo de los cuales la exposición durante toda la vida o por un tiempo promedio determinado no constituye un riesgo significativo para la salud, pero si estos límites son sobrepasados en el corto plazo, ello no implica que se producirán efectos adversos inmediatamente sino que se incrementarán las posibilidades de que dichos efectos de produzcan.

Los contaminantes del aire en interiores generalmente se diferencian de los del aire en exteriores por el tipo y nivel de concentración.

En las guías para la calidad del aire basadas en la salud, se describen brevemente los contaminantes clave del aire, también denominados “clásicos” (CO, N₂, CO₂, O₃, SO₂).

Según la Dirección General de Investigación y Asuntos Ambientales del Servicio Nacional de Meteorología e Hidrología y la Dirección General De Salud Ambiental con Decreto Supremo 009-2003-SA y el Decreto Supremo 012-2005-SA nos proveen de la clasificación de estados de la calidad del aire, a continuación se presenta un cuadro en detalle.

2.2.5.5 CLASIFICACIÓN DE LOS ESTADOS DE LA CALIDAD DEL AIRE

ESTADO DE CALIDAD DEL AIRE	NIVEL DE CONCENTRACIÓN	EFFECTOS EN LA SALUD
BUENO	0 a 100 ppm	No hay riesgos en la salud.
MODERADO	>100 a 180 ppm	Las personas sensibles pueden presentar alergias, tos, cansancio.
MALO	>180 a 240 ppm	Las personas pueden padecer síntomas como tos seca, ojos cansados, dolor de cabeza, ardor en la nariz y garganta, mareos.
MUY MALO	>240 a 350 ppm	Dolor de cabeza, confusión mental, náuseas, desmayos, puede ocasionar la muerte si estos niveles, aunque moderados, se respiran por mucho tiempo
ALERTA MAXIMA	>350 a 550 ppm	La muerte en pocos segundos

TABLA 2.5 CLASIFICACIÓN DE LOS ESTADOS DE LA CALIDAD DEL AIRE

2.2.5.6 SENSORES DE GAS

Un sensor es un dispositivo para detectar y señalar una condición de cambio, tales como magnitudes físicas o químicas y transformarlas en variables eléctricas. Los sensores posibilitan la comunicación entre el mundo físico y los sistemas de medición y/o control, tanto eléctricos como electrónicos, utilizándose extensivamente en todo tipo de procesos para propósitos de monitoreo, medición, control y procesamiento.

Para el diseño de este proyecto se utilizará un sensor de gas. Los sensores de gases generalmente constan de elementos físicos sobre los cuales se deposita una capa activa capaz de absorber y/o interactuar con ciertos gases. Estas capas actúan como elementos detectores de gases y dan una gran información de los elementos químicos que conforman el medio ambiente.

2.2.5.6.1 SENSOR DE GAS MQ-2

Este sensor se utiliza en la detección de monóxido de carbono (CO) en la industria de automóviles, también son adecuados para la detección de GLP, I-butano, gas propano, metano, alcohol, hidrogeno y humo.

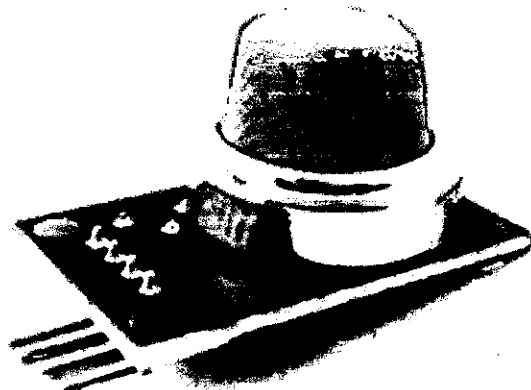


FIG. 2.16 SENSOR DE GAS MQ-2

2.2.5.6.2 CARACTERÍSTICAS

- Detección de gran alcance
- Rápida respuesta y alta sensibilidad
- Estable y de larga vida
- Circuito de accionamiento sencillo

2.2.5.6.3 ESPECIFICACIONES

2.2.5.6.3.1 CONDICIONES DE TRABAJO

SÍMBOLO	NOMBRE DE PARÁMETRO	CONDICIONES TECNICAS	OBSERVACIONES
Vcc	Voltaje del circuito	5V±0.1	AC ó DC
VH	Voltaje de calentamiento	5V±0.1	AC ó DC
RL	Resistencia de carga	Poder de ajuste	
RH	Resistencia de calentamiento	33Ω±5%	Pieza TEM*
PH	Calentamiento por consumo	Menos que 800mw	

TABLA 2.6 CONDICIONES DE TRABAJO DEL SENSOR MQ2

*TEM (TRANSVERSAL ELECTRO-MAGNETICA)

2.2.5.6.3.2 CONDICIÓN AMBIENTAL

SIMBOLO	NOMBRE DE PARÁMETRO	CONDICIONES TECNICAS	OBSERVACIONES
Tao	Usando TEM	-20°C-50°C	
Tas	Almacenamiento en	-20°C-70°C	
RH	Humedad relativa	Menos que el 95%Rh	
O2	Concentración de oxígeno	21%(norma de estado) La concentración del oxígeno puede afectar la sensibilidad	Por encima del valor mínimo 2%

TABLA 2.7 CONDICIÓN AMBIENTAL DEL SENSOR MQ2

2.2.5.6.3.3 CARACTERÍSTICAS DE SENSIBILIDAD

SIMBOLO	NOMBRE DE PARAMETRO	CONDICIONES TECNICAS	OBSERVACIONES
Rs	Resistencia de sensibilidad	3KΩ-30KΩ (1000ppm iso-butano)	Detección de la concentración de alcance : 200ppm-5000ppm GLP y propano 300ppm-5000ppm butano 5000ppm-20000ppm metano 300ppm-5000ppm H2 100ppm-2000ppm Alcohol
α (3000/1000) Iso-butano	Tasa de concentración	≤ 0.6	
Estándar y condición de detección	Temperatura: 20°C±2°C Vc:5V±0.1 Humedad: 65%±5% Vh: 5V±0.1		
Tiempo de recalentamiento	Sobre las 24 horas		

TABLA 2.8 CARACTERÍSTICAS DE SENSIBILIDAD

Este sensor es de alta sensibilidad y con un tiempo de respuesta rápido, consta de 4 pines macho para su conexión (pin de alimentación, pin de tierra, pin de salida analógica y pin de salida digital).

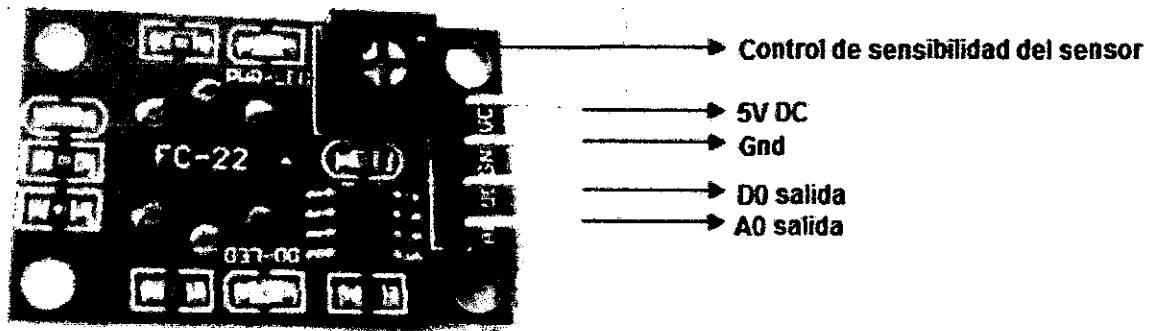


FIG. 2.17 DESCRIPCION DE CADA PIN

Las dimensiones del módulo del sensor es: 32mmm X 22mm X 24mm

La unidad de medida de los sensores MQ-2 es ppm (Partes por millón).

2.2.6. SISTEMA OPERATIVO ANDROID

2.2.6.1 ¿QUÉ ES ANDROID?

Android es un sistema operativo para dispositivos móviles como teléfonos inteligentes y tabletas basado en el núcleo Linux. Es desarrollado por la Open Handset Alliance, la cual es liderada por Google, usando diversos conjuntos de herramientas de software de código abierto para dispositivos móviles. Fue construido para permitir a los desarrolladores la creación de aplicaciones móviles que aprovechan al máximo el uso de todas las herramientas que un dispositivo como este puede ofrecer.

Implementa una arquitectura en la que cualquier aplicación puede obtener acceso a las capacidades del teléfono móvil. Por ejemplo, una aplicación puede llamar una o varias de las funcionalidades básicas de los dispositivos móviles, tales como realizar llamadas, enviar mensajes de texto, o utilizar la cámara, facilitando a los desarrolladores crear experiencias más ricas y con más coherencia para los usuarios.

Está construido sobre el kernel de Linux. Además, se utiliza una máquina personalizada virtual que fue diseñada para optimizar los recursos de memoria y de hardware en un entorno móvil. Android es de código abierto, y además puede ser libremente ampliado para incorporar nuevas tecnologías de vanguardia que van surgiendo. La plataforma continuará evolucionando a medida que la comunidad de desarrolladores trabajando juntos puedan crear aplicaciones móviles innovadoras.

2.2.6.2 HISTORIA DE ANDROID

Fue desarrollado por Android Inc., empresa que en 2005 fue comprada por Google, aunque no fue hasta 2008 cuando se popularizó, gracias a la unión al proyecto de Open Handset Alliance, un consorcio formado por 48 empresas de desarrollo hardware, software y telecomunicaciones, que decidieron promocionar el software libre. Pero ha sido Google quien ha publicado la mayor parte del código fuente del sistema operativo, gracias al software Apache, que es una fundación que da soporte a proyectos software de código abierto.

Dado que Android está basado en el núcleo de Linux, tiene acceso a sus recursos, pudiendo gestionarlo, gracias a que se encuentra en una capa por encima del Kernel, accediendo así a recursos como los controladores de pantalla, cámara, memoria flash.

2.2.6.3 VERSIONES DISPONIBLES

El sistema operativo Android, al igual que los propios teléfonos móviles, ha evolucionado rápidamente, acumulando una gran cantidad de versiones, desde la 1.0 para el QWERTY HTC G1, hasta la 4.4.2 que acaba de salir al mercado.

- CUPCAKE: ANDROID VERSIÓN 1.5

Características: Widgets, teclado QWERTY virtual, copy & paste, captura de vídeos y poder subirlos a Youtube directamente.

- DONUT: ANDROID VERSIÓN 1.6

Características: Añade a la anterior la mejora de la interfaz de la cámara, búsqueda por voz, y navegación en Google Maps.

- ECLAIR: ANDROID VERSIÓN 2.0/2.1

Características: Mejoras en Google Maps, salvapantallas animado, incluye zoom digital para la cámara, y un nuevo navegador de internet.

- FROYO: ANDROID VERSIÓN 2.2

Características: Incluye hotspot Wifi, mejora de la memoria, más veloz, Microsoft Exchange y video-llamada.

- GINGER BREAD: ANDROID VERSION 2.3

Características: Mejoras del consumo de batería, el soporte de vídeo online y el teclado virtual, e incluye soporte para pagos mediante NFC.

- **HONEY COMB: ANDROID VERSION 3.0/3.4**

Características: Mejoras para tablets, soporte Flash y Divx, integra Dolphin, multitarea pudiendo cambiar de aplicación dejando las demás en espera en una columna, widgets y homepage personalizable.

- **ICE CREAM SANDWICH: ANDROID VERSION 4.0**

Características: Multiplataforma (tablets, teléfonos móviles y netbooks), barras de estado, pantalla principal con soporte para 3D, widgets redimensionables, soporte usb para teclados, reconocimiento facial y controles para PS3.

2.2.6.2 ARQUITECTURA DE LA PLATAFORMA ANDROID

La arquitectura interna de la plataforma Android, está básicamente formada por 4 componentes: aplicaciones, almacén de aplicaciones, librerías y kernel/Linux. En la Figura 2.18, se muestran las capas que conforman el sistema operativo Android.

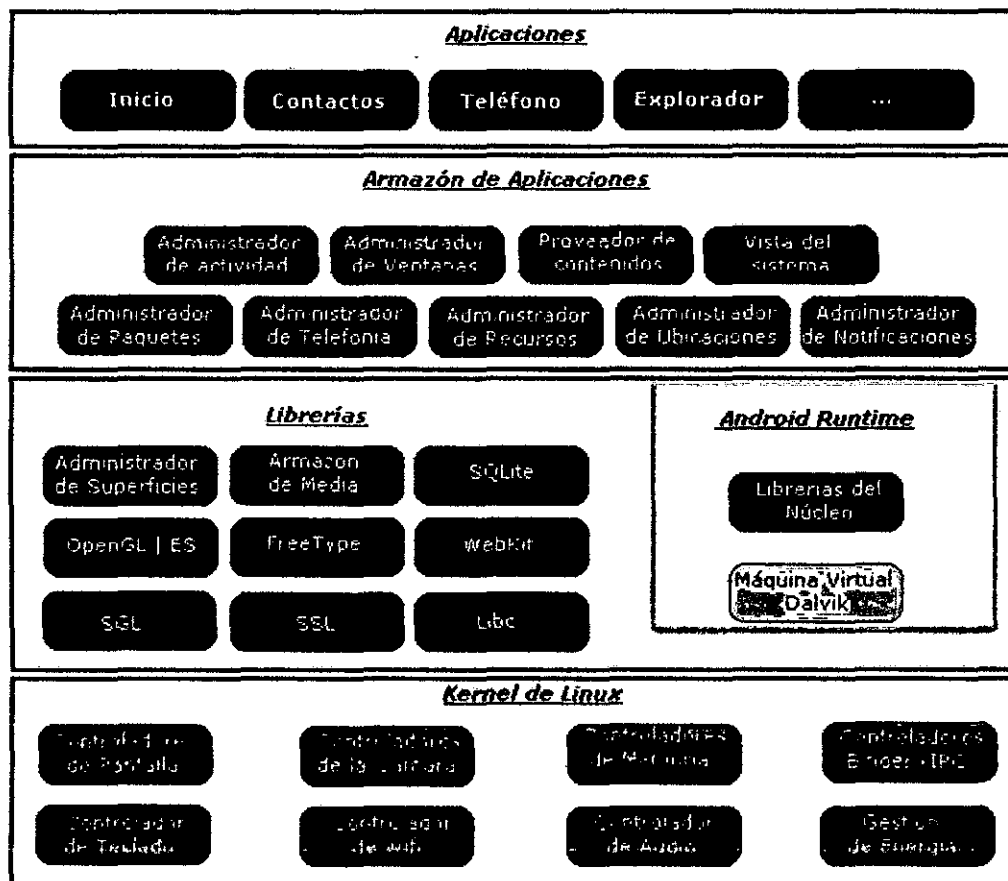


FIG. 2.18 SISTEMAS DE CAPAS DE ANDROID

2.2.6.2.1 KERNEL

Un núcleo o kernel es un software que constituye la parte más importante del sistema operativo. Es el principal responsable de facilitar a los distintos programas acceso seguro al hardware de la computadora o en forma básica, es el encargado de gestionar recursos, a través de servicios de llamada al sistema.

Como hay muchos programas y el acceso al hardware es limitado, también se encarga de decidir qué programa podrá hacer uso de un dispositivo de hardware y durante cuánto tiempo, lo que se conoce como multiplexado. Acceder al hardware directamente puede ser realmente complejo, por lo que los núcleos suelen implementar una serie de abstracciones del hardware. Esto permite esconder la complejidad, y proporciona una interfaz limpia y uniforme al hardware subyacente, lo que facilita su uso al programador.

El kernel lo podemos definir como el corazón del SO. En esta serie los modelos de desarrollo han cambiado, la manera de numerarse es de 4 dígitos (VV.RR.NR.CR).

VV: Indica la versión (o serie) del kernel.

RR: Indica la revisión del kernel (Da igual que los vea de forma impar o par, hoy en día, no tiene significado)

NR: Indica nuevas revisiones del kernel. Estos números cambian cuando se incorporan nuevas características y drivers

CR: Este dígito cambia cuando se corrigen fallos de programación o fallos de seguridad dentro de una revisión.

Android utiliza el núcleo de Linux 2.6 como una capa de abstracción para el hardware disponible en los dispositivos móviles. Esta capa contiene los drivers necesarios para que cualquier componente hardware pueda ser utilizado mediante las llamadas correspondientes. Siempre que un fabricante incluye un nuevo elemento de hardware, lo primero que se debe realizar para que pueda ser utilizado desde Android es crear las librerías de control o drivers necesarios dentro de este kernel de Linux embebido en el propio Android.

2.2.6.3 GUÍA PARA DESARROLLADORES (BÁSICA)

La Guía ofrece una introducción práctica a desarrollar aplicaciones para Android y documentación sobre las características de las plataformas más importantes. Se estudian los conceptos detrás de Android, el marco para la construcción de una aplicación y las herramientas para el desarrollo, la prueba y la publicación de software para la plataforma.

La Guía del desarrollador tiene la mayor parte de la documentación para la plataforma Android, con excepción de material de referencia en el Framework de la API, para conocer las especificaciones de la API.

ANDROID BASICS

Es una orientación inicial para Android, lo que es, lo que ofrece y cómo se ajusta su aplicación.

TEMAS DEL FRAMEWORK

Son las discusiones sobre determinadas partes del framework de Android y la API. Para una introducción al marco, comienzan con Application Fundamentals; luego explora otros temas, desde el diseño de una interfaz de usuario y la creación de recursos para el almacenamiento de datos; y el uso de los permisos.

TEMAS DEL ANDROID MARKET

La documentación de los temas que conciernen a la publicación y monetización de las aplicaciones en AndroidMarket, por ejemplo, cómo hacer cumplir las políticas de concesión de licencias y ponerlo en práctica en la aplicación de facturación.

DESARROLLO

Indicación es para el uso de desarrollo de Android y herramientas de depuración, y para comprobar los resultados.

PUBLICACIÓN

Las instrucciones sobre cómo preparar su aplicación para la implementación y la forma de publicar cuando esté listo.

APLICACIONES WEB

Hace referencia a la documentación sobre cómo crear aplicaciones web que funcionan perfectamente en dispositivos con Android y como crear aplicaciones Android para incrustar contenido basado en web.

APÉNDICE

Información de referencia y especificaciones, así como preguntas frecuentes, un glosario de términos, y otra información. El primer paso en la programación para Android es la descarga del SDK (software development kit). Después de tener el SDK, se empieza por buscar a través de the Dev Guide. Si se quiere empezar por conseguir un rápido vistazo a algo de código, el tutorial Hola Mundo los lleva a través de una aplicación "Hello World" para introducir algunos conceptos básicos de una aplicación Android.

2.2.6.4 PAUTAS PARA LAS INTERFACES DE USUARIO

El sistema operativo Android fue adoptado con gran rapidez por múltiples fabricantes de dispositivos móviles, que lo adoptaron como plataforma debido a su carácter más abierto.

Esto potenció a Android, y su tienda de aplicaciones (el AndroidMarket) comenzó a contar con una cantidad enorme de desarrollos (a la fecha, más de 400.000). Sin embargo, esto mismo generó en Android una falta de estandarización en las diferentes aplicaciones, que ha generado también ligeras incompatibilidades entre las distintas implementaciones del sistema operativo de los fabricantes.

Para la última versión mayor de Android (4.0) Google comenzó a tomar algunas cartas en el asunto, y para ello liberó una página llamada “AndroidDesign” que define una serie de principios para el diseño de interfaces y afirman:

“Estos principios de diseño fueron desarrollados por y para el Equipo de Experiencia de Usuario de Android teniendo en mente las mejores intenciones y consideraciones. Se deben tener en cuenta a la hora de aplicar las ideas creativas de diseño.” [9]

versión 4.0 de Android, pocas aplicaciones del AndroidMarket las cumplen, y en este momento no son un requerimiento para conseguir la aprobación en la tienda oficial de Android. La página de “AndroidDesing” inicia planteando una serie de principios generales para el diseño de interfaces gráficas.

Las aplicaciones nativas de Android siguen tres principios generales:

Encántame: Las aplicaciones deben combinar belleza, simplicidad y propósito, para crear experiencias poderosas y de mínima dificultad de uso.

Simplifica mi vida: Las aplicaciones de Android deben facilitar la vida y ser fáciles de entender. Cuando las personas utilicen por primera vez una aplicación, deben deducir de manera intuitiva las características más importantes.

Sorpréndeme: No es suficiente el hacer una aplicación fácil de utilizar. Las aplicaciones de Android deben empoderar a las personas a intentar nuevas cosas y a usarlas de manera creativa. Todos estos principios generales de diseño se traducen en una serie de elementos más concretos agrupados en tres grandes grupos: Estilo, patrones y bloques de construcción.

[9] Android Developers. Design. [En línea]. <http://developer.android.com/guide/index.html>. [Citado el 16 de agosto del 2014]

2.2.6.5 APP INVENTOR

App Inventor es una aplicación originalmente desarrollada por Google y mantenida ahora por el Instituto de Tecnología de Massachusetts. Permite que cualquier persona, incluyendo las no familiarizadas con la programación y SDK de Android, pueda crear aplicaciones de Software para Android. Utiliza una interfaz gráfica, muy similar al Scratch y el StarLogo, que permite a los usuarios arrastrar y soltar objetos visuales para crear una aplicación que puede ejecutarse en el sistema Android.

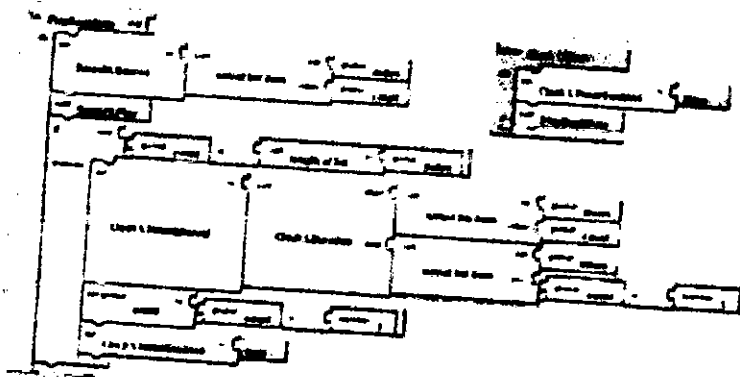


FIG. 2.19 ENTORNO DE DESARROLLO DE APP INVENTOR

Google puso fin al desarrollo el 31 de diciembre de 2011 cediéndole el código al MIT, quien lo ha puesto a disposición de todos. Se trata de una utilidad Web desarrollada por Google que permite realizar aplicaciones para Android sin escribir código Java, todo de forma visual e intuitiva (uniendo piezas de un puzle).

Una característica interesante es que el desarrollo de la aplicación es en Web. Aunque es necesario instalar un módulo de software en la computadora, en el momento del desarrollo se ejecuta la última versión del App Inventor disponible en su sitio web y los proyectos se guardan en línea. El App Inventor consta de dos segmentos principales:

- Un módulo Web y
- El editor de bloques de Android.

El módulo en Web que se mencionaba, donde aparte de ser el punto de entrada tenemos acceso a nuestros proyectos y, una vez abierto un proyecto, podemos entrar a la sección de diseño de nuestra aplicación. Esta sección es donde podemos añadir los componentes y configurarlos apropiadamente. Si se trata de componentes visuales, entonces definimos también el diseño de la interfaz. Para los familiarizados con desarrollo de aplicaciones mediante componentes visuales verán que es un concepto bastante similar.

El segmento del editor de bloques se verá más adelante, por el momento basta con mencionar que ahí es donde los bloques se conectan cual piezas Lego, formando la lógica de la aplicación.

Permite a cualquiera crear aplicaciones de software para el sistema operativo Android.

Utiliza una interfaz gráfica que permite a los usuarios arrastrar y soltar objetos visuales para crear una aplicación que puede ejecutarse en el sistema Android, que funciona en muchos dispositivos móviles.

Todo ello sin usar ni una sola línea de código, de forma intuitiva y gráfica. La aplicación se puso a disposición de los usuarios, mediante invitación, el 12 de julio de 2010, el 15 de diciembre de 2010 se puso a disposición de usuarios registrados. La aplicación está dirigida a personas que no están familiarizadas con la programación de computadoras. La idea es que cualquier persona pueda desarrollarse sus propias aplicaciones para su dispositivo Android.

2.2.6.5.1 ¿PORQUÉ APP INVENTOR?

Porque es gratuito y poco exigente con los requisitos técnicos, funciona online y sin apenas instalación. Ha sido probado por estudiantes con excelentes resultados en institutos de enseñanza de San Francisco, eso es punto de partida interesante.

El lenguaje de programación que se usa en App Inventor es bastante similar al de Scratch, también desarrollado en el MIT. Ya hemos trabajado un poco con Scratch durante este curso. Cuando conocimos App Inventor no tuvimos ninguna duda. Había que probarlo.

2.2.6.5.2 REQUERIMIENTOS DEL SISTEMA

COMPUTADORA Y SISTEMA OPERATIVO

- **Macintosh** (with Intel processor): Mac OS X 10.5, 10.6.
- **Windows**: Windows XP, Windows Vista, Windows 7.
- **GNU/Linux**: Ubuntu 8+, Debian 5+.

NAVEGADOR WEB

- **Mozilla Firefox 3.6** o superior.
- **Apple Safari 5.0** o superior.
- **Google Chrome 4.0** o superior.
- **Microsoft Internet Explorer 7** o superior.

Para poder acceder al App Inventor se debe haber instalado Java 6 o superior en el ordenador o PC ya que no funcionará en el equipo, lo recomendable es no tratar de utilizar la aplicación App Inventor sin haberlo instalado.

Luego de ello se debe crear una cuenta en Gmail para poder tener un usuario y clave para acceder al App Inventor. La interfaz de esta herramienta se muestra a continuación:

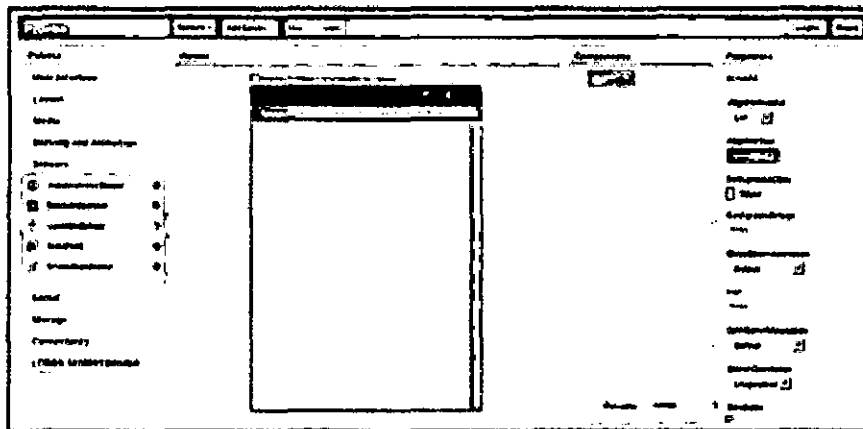






FIG. 2.20 INTERFAZ DE APP INVENTOR 2

Para realizar pruebas de la aplicación creada tenemos:

- 

 Si se está utilizando un dispositivo Android y se tiene un cable USB, se puede enviar la aplicación al dispositivo Android e instalarlo.
- 

 Se puede hacer uso de un emulador.

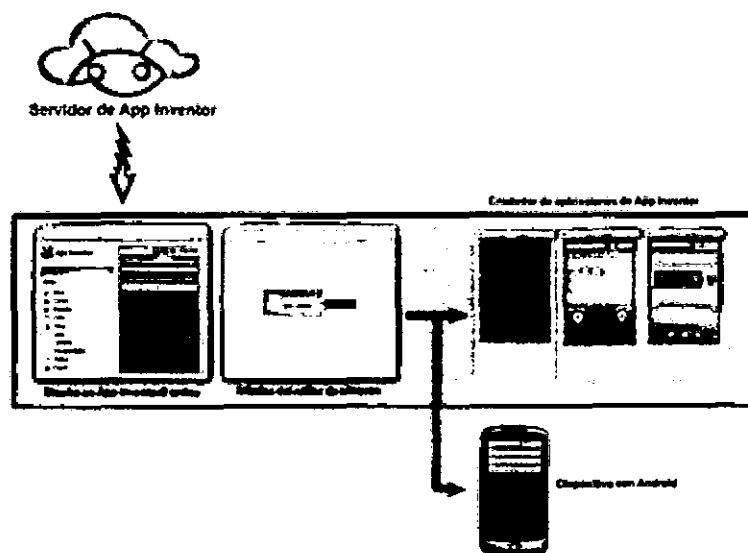


FIG. 2.21 USO DE EMULADOR VIRTUAL PARA PRUEBA DE LA APLICACIÓN

CAPITULO III

3.0 CONSTRUCCIÓN DE LA APLICACIÓN

3.1 ESTUDIO DE FACTIBILIDAD

Como se verá en el desarrollo de este trabajo los mecanismos y componentes necesarios para su implementación se encuentran en el mercado y en tiendas online a precios accesibles. El hardware lo componen el sensor de gas, el microcontrolador, el dispositivo móvil (con S. O. Android), y variados componentes electrónicos, y en la parte virtual utilizamos software de programación online y accesible también: Pic Basic Pro y APP Inventor2, además del software Pickit2 que ello conlleva a conocimientos e investigación en este ámbito.

Por lo tanto en términos de factibilidad el proyecto es viable.

3.2 ANÁLISIS DEL SISTEMA

3.2.1 DESCRIPCIÓN DE LAS ACTIVIDADES

Se presenta el siguiente cronograma de actividades para el proyecto:

ACTIVIDAD	1	2	3	4	5	6
Elaboración del Proyecto	X					
Recepción de Información		X				
Diseño del Hardware			X	X		
Diseño del Software					X	
Pruebas						X
Elaboración de Informe Final						X

TABLA 3.0 DESCRIPCIÓN DE ACTIVIDADES

3.3 DISEÑO DEL SISTEMA

3.3.2 DISEÑO DEL HARDWARE DEL SISTEMA

3.3.2.1 DIAGRAMA DE BLOQUES DEL SISTEMA DISEÑADO

En el diagrama de bloques se puede apreciar que los leds y buzzer son los dispositivos de salida los cuales están conectados al puerto B (b.4, b.5, b.6, y b.7 respectivamente), definidos como alarmas del sistema cuando excede el nivel de lectura definido en el microcontrolador como se verá más adelante en el circuito propuesto, el modulo sensor de gas es un dispositivo de entrada conectado en el puerto analógico (a.0), y el módulo bluetooth está conectado al puerto B (b.0 y b.1) encargado de transmitir la lectura del módulo sensor al visualizador es decir al dispositivo celular el cual posee una aplicación en formato .apk programado en android (servidor de google página: web de App Inventor2).

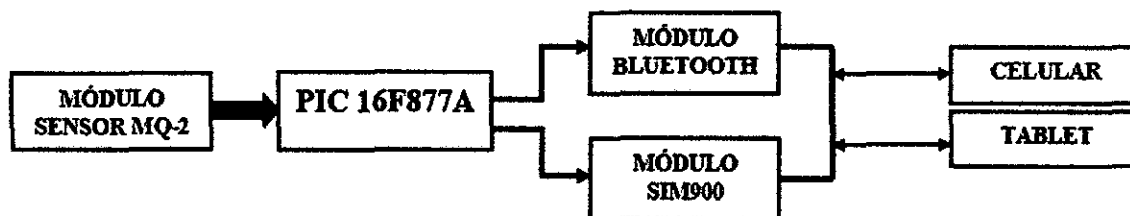


FIG. 3.0 DIAGRAMA DE BLOQUES DEL SISTEMA

3.3.2.2 ESQUEMA DEL CIRCUITO EN EAGLE

El siguiente esquema muestra la estructura que compone al sistema desarrollado y muestra las conexiones de los componentes utilizados.

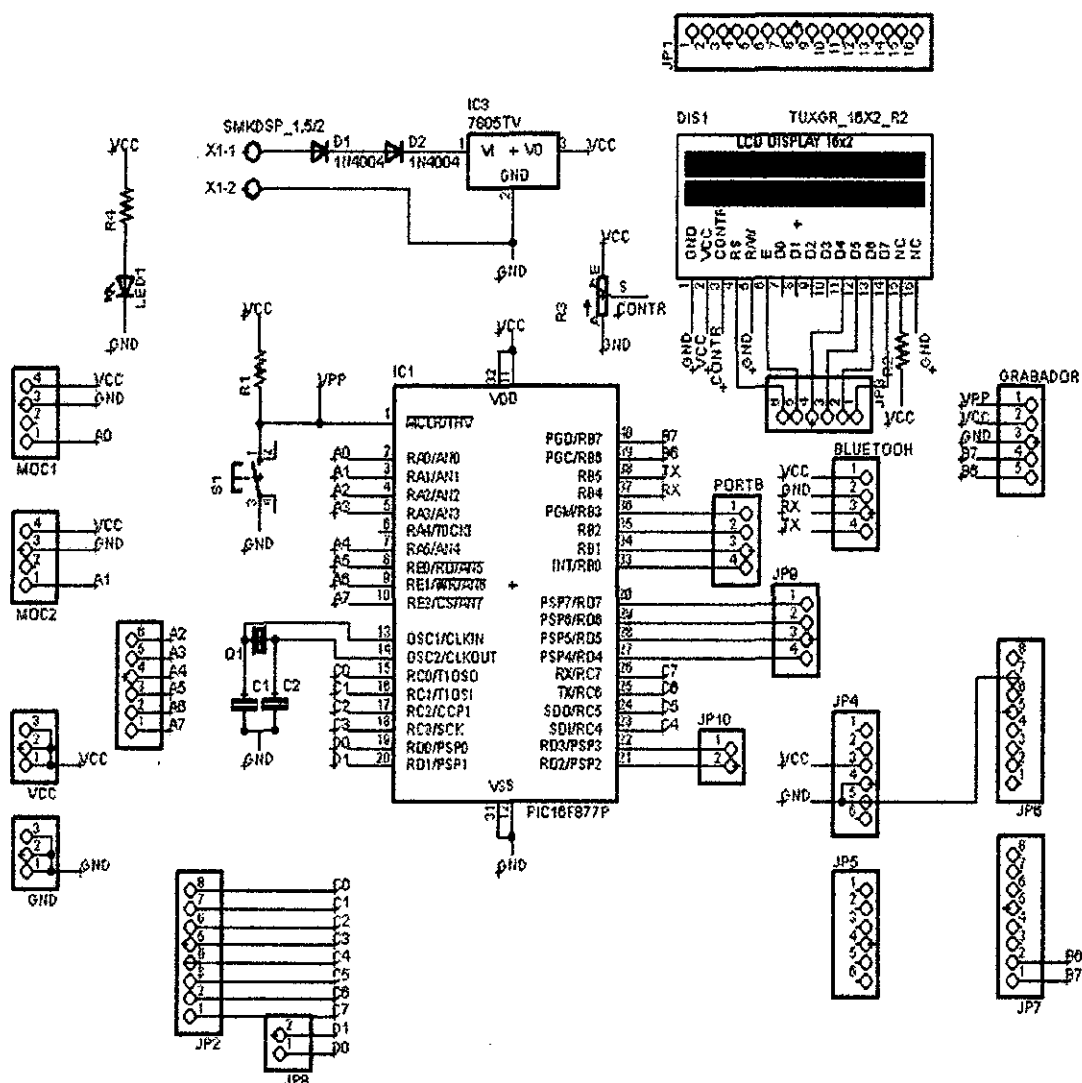


FIG. 3.1 CONEXIONES DEL SISTEMA IMPLEMENTADO

El siguiente esquema muestra el ruteo de la placa en Eagle

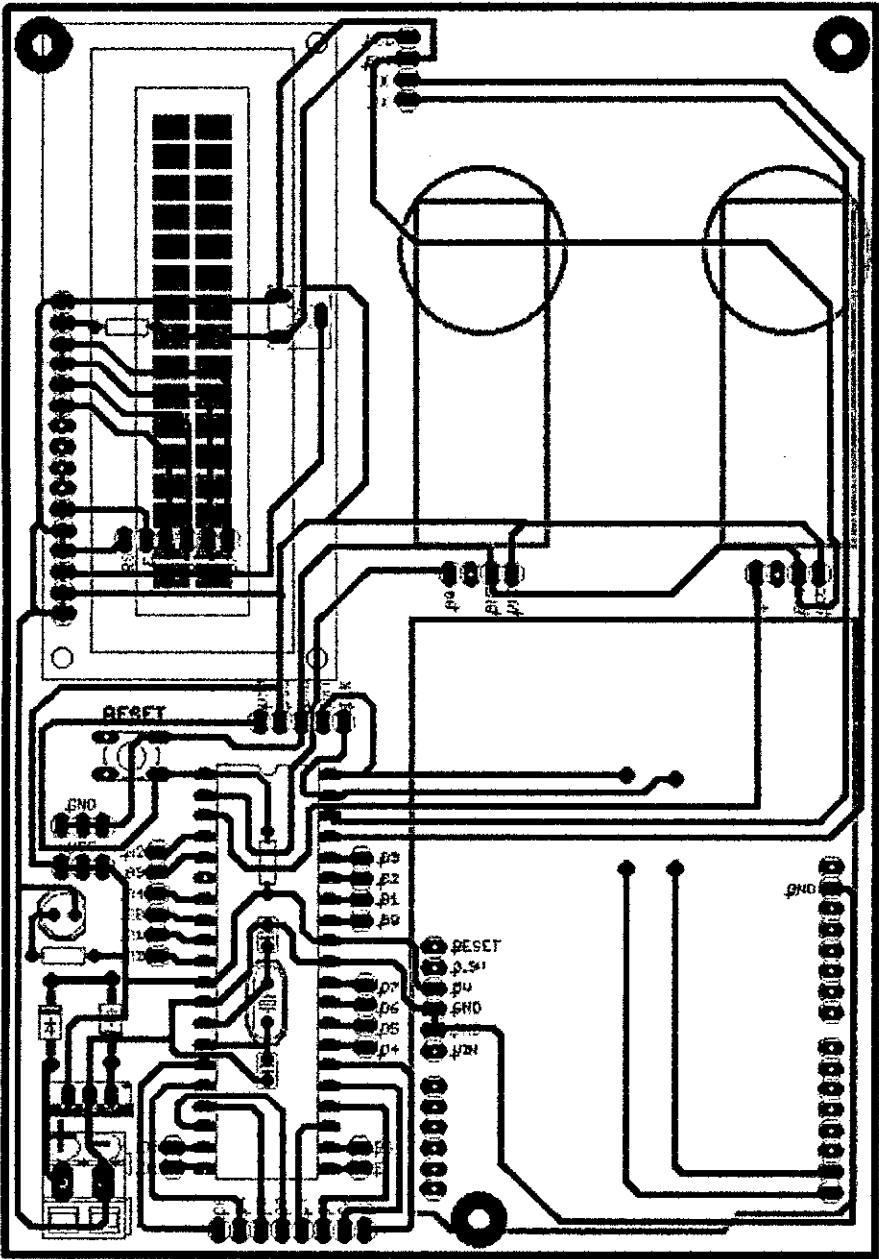


FIG. 3.2 RUTEO DEL CIRCUITO

3.3.3 DISEÑO DEL SOFTWARE DEL SISTEMA

3.3.3.1 PROGRAMACIÓN EN PIC BASIC PRO DEL MICROCONTROLADOR PIC 16F877A

DEFINE OSC 4

'DEFINICIONES PARA LCD

Define LCD_DREG PORTD

Define LCD_DBIT 4

Define LCD_RSREG PORTD

Define LCD_RSBIT 3

Define LCD_EREG PORTD

Define LCD_EBIT 2

'CARACTERISTICAS DEL ADC

DEFINE ADC_BITS 8

DEFINE ADC_CLOCK 3

DEFINE ADC_SAMPLEUS 50

TRISA = %11111111' Set PORTA to all input

ADCON1 = %00000000 ' Set PORTA analog

'VARIABLES DEL ADC

ch0 var byte 'CO1

ch1 var byte 'CO2

canal0 VAR WORD

canal1 VAR WORD

'PUERTOS PARA COMUNICACION CON MODEM GSM

TX VAR PORTB.6:RX VAR PORTB.7

modem VAR BYTE [20]

D0 VAR BYTE:D1 VAR BYTE:D2 VAR BYTE:D3 VAR BYTE

D4 VAR BYTE:D5 VAR BYTE:D6 VAR BYTE:D7 VAR BYTE

D8 VAR BYTE:D9 VAR BYTE:D10 VAR BYTE

VALOR VAR BYTE:TEMP1 VAR word:TEMP2 VAR word

TEMP3 VAR word:TEMP4 VAR word:clave var word

clavel var word

DATOS VAR BYTE [11]

numero1 var byte[9]:numero2 var byte[9]

numero3 var byte[9]:numero4 var byte[9]

A VAR Byte

read 50,A

```

REGISTRAR:
LCDOut $fe, 1
LCDOut "REGISTRANDO..."
SerOut2 TX,188,["AT+CGREG?",13] ' PREGUNTA SI MODEN ESTA REGISTRADO CON
ALGUNA OPERADORA
SerIn2 RX,188,1000,REGISTRAR,[WAIT ("+CGREG:"),STR modem \4]
Pause 100
IF MODEM[3]="1" Then
    LCDOut $fe, 1
    LCDOut $fe, 2
    LCDOut "MODEM REGISTRADO"
    Pause 1000
Else
    GoTo REGISTRAR
EndIF
*****
GOSUB LEER_NUM
*****
PANTALLA INICIAL
LCDOut $fe, 1
LCDOut " MEDIDOR DE GAS"
LCDOut $fe,$C0," UNP--2015"
PAUSE 1000
*****
Inicio:
SerIn2 RX,188,3000,alarma,[wait("R"),STR DATOS \11]
    D0=DATOS[0]
    D1=DATOS[1]
    D2=DATOS[2]
    D3=DATOS[3]
    D4=DATOS[4]
    D5=DATOS[5]
    D6=DATOS[6]
    D7=DATOS[7]
    D8=DATOS[8]
    D9=DATOS[9]
    D10=DATOS[10]
*****
    LCDOut $fe, 1
    LCDOut "Msg. Recibido!!!"
    LCDOut $fe,$C0,D0,D1,D2,D3,D4,D5,D6,D7,D8,D9,D10
    PAUSE 2000
    LCDOut $fe, 1

```

'SELECCION DE CASOS

SELECT CASE D0

'LEER ESTADO DE LOS SENSORES

case "V" 'comando es: 'RVccccXXXXXX, CCCC=CLAVE (RV1234XXXXX)

TEMP1=d1-48

TEMP1=TEMP1*1000

TEMP2=d2-48

TEMP2=TEMP2*100

TEMP3=d3-48

TEMP3=TEMP3*10

TEMP4=d4-48

CLAVE1=TEMP1+TEMP2+TEMP3+TEMP4 'SE OBTIENE CLAVE

if clave=clavel then

'LEER CANALES ANALOGICOS

ADCIN 0, ch0 'co1

ADCIN 1, ch1 'co2

PAUSE 50

GOSUB enviar_valor

LCDOut \$fe, 1

LCDOut "Msg. Enviado!!!"

pause 1000

endif

'PROGRAMAS NUMERO DE CELULAR QUE RECIBIRAN ESTADOS DE ALARMA

CASE "T" 'RTd1d2d3d4d5d6d7d8d9d10 ejemp: RT#968887721 comando =(RT1951551591)

numero4[0]=d2-48

numero4[1]=d3-48

numero4[2]=d4-48

numero4[3]=d5-48

numero4[4]=d6-48

numero4[5]=d7-48

numero4[6]=d8-48

numero4[7]=d9-48

numero4[8]=d10-48

Select Case D1

Case "1" ' se programa telefono 1

write 0,numero4[0]

write 1,numero4[1]

write 2,numero4[2]

write 3,numero4[3]

write 4,numero4[4]

write 5,numero4[5]

```

write 6,numero4[6]
write 7,numero4[7]
write 8,numero4[8]

```

'GoSub Prog_num1

Case "2" ' se programa telefono 2

```

write 9,numero4[0]
write 10,numero4[1]
write 11,numero4[2]
write 12,numero4[3]
write 13,numero4[4]
write 14,numero4[5]
write 15,numero4[6]
write 16,numero4[7]
write 17,numero4[8]

```

'GoSub Prog_num2

Case "3" ' se programa telefono 3

```

write 18,numero4[0]
write 19,numero4[1]
write 20,numero4[2]
write 21,numero4[3]
write 22,numero4[4]
write 23,numero4[5]
write 24,numero4[6]
write 25,numero4[7]
write 26,numero4[8]

```

'GoSub Prog_num3

End Select

GOSUB LEER_NUM

'PROGRAMAR CLAVE

CASE "P" 'RPnnnnXXXXXX COMANDO donde nn es un numero entre 0000 y 9999

```

modem[4]=d1-48
modem[5]=d2-48
modem[6]=d3-48
modem[7]=d4-48
write 36,modem[4]
write 37,modem[5]
write 38,modem[6]
write 39,modem[7]

```

GOSUB LEER_NUM

CASE "A" 'ARMAR ALARMA PARA ENVIO DE SMS DE ALERTA, comando es:
'RAccccXXXXXX

```
TEMP1=d1-48
TEMP1=TEMP1*1000
TEMP2=d2-48
TEMP2=TEMP2*100
TEMP3=d3-48
TEMP3=TEMP3*10
TEMP4=d4-48
CLAVE1=TEMP1+TEMP2+TEMP3+TEMP4 'SE OBTIENE CLAVE
if clave=clave1 then
    A=0
    write 50,0
ENDIF
end select
GoTo Inicio
```

alarma:

'LEER CANALES ANALOGICOS

```
ADCIN 0, ch0 'co1
ADCIN 1, ch1 'co2
PAUSE 50
```

'VIZUALIZAR LOS VALORES DE LAS VARIABLES MEDIDAS

```
LCDOut $fe, 1
LCDOut "MEDIDOR DE GAS"
LCDOut $fe,$C0,"C1=",DEC CH0," C2=", DEC CH1
PAUSE 100
```

**'ACONDICIONAR LOS VALORES DE LAS VARIABLES PARA QUE CADENA DE DATOS
SEA DE UN MISMO TAMAÑO SIEMPRE**

```
canal0=ch0+1000
canal1=ch1+1000
```

'ENVIAR CADENA DE DATOS SERIALMENTE Y POR BLUETOOTH HC06

```
SEROUT2 PORTB.5,84, ["A",DEC canal0,"B",dec canal1,10,13]
pause 50
IF CH0>700 OR CH1 >700 THEN
    GOSUB MENSAJE_AVISO
ENDIF
```

goto Inicio ' SALTA A ETIQUETA INICIO PARA INICIAR TODO DE NUEVO

'LEER NUMERO DE CELULARES Y CLAVE

LEER_NUM:

```
LCDOut $fe, 1
LCDOut "LEER NUMEROS..."
```

```

Pause 100
read 0,numero1[0]
read 1,numero1[1]
read 2,numero1[2]
read 3,numero1[3]
read 4,numero1[4]
read 5,numero1[5]
read 6,numero1[6]
read 7,numero1[7]
read 8,numero1[8]
read 9,numero2[0]
read 10,numero2[1]
read 11,numero2[2]
read 12,numero2[3]
read 13,numero2[4]
read 14,numero2[5]
read 15,numero2[6]
read 16,numero2[7]
read 17,numero2[8]
read 18,numero3[0]
read 19,numero3[1]
read 20,numero3[2]
read 21,numero3[3]
read 22,numero3[4]
read 23,numero3[5]
read 24,numero3[6]
read 25,numero3[7]
read 26,numero3[8]
read 36,modem[4]
read 37,modem[5]
read 38,modem[6]
read 39,modem[7]
'CLAVE
TEMP1=MODEM[4]
TEMP1=TEMP1*1000
TEMP2=MODEM[5]
TEMP2=TEMP2*100
TEMP3=MODEM[6]
TEMP3=TEMP3*10
TEMP4=MODEM[7]
CLAVE=TEMP1+TEMP2+TEMP3+TEMP4 'SE OBTIENE CLAVE PARA ARMAR ALARMA
LCDOut $fe, 1
LCDOut " Tel:",dec numero1[0],dec numero1[1],dec numero1[2],dec numero1[3],dec
numero1[4],dec numero1[5],dec numero1[6],dec numero1[7],dec numero1[8]
Pause 1000
LCDOut $fe, 1

```

```

LCDOut " Te2:",dec numero2[0],dec numero2[1],dec numero2[2],dec numero2[3],dec
numero2[4],dec numero2[5],dec numero2[6],dec numero2[7],dec numero2[8]
Pause 1000
LCDOut $fe, 1
LCDOut " Te3:",dec numero3[0],dec numero3[1],dec numero3[2],dec numero3[3],dec
numero3[4],dec numero3[5],dec numero3[6],dec numero3[7],dec numero3[8]
Pause 1000
LCDOut $fe, 1
LCDOut " LA CLAVE ES:"
LCDOUT $FE,$C0,dec4 clave
PAUSE 1000
LCDOut $fe, 1
RETURN
enviar_valor:
    serout2 TX,188,["at+cmgs=",34,dec numero1[0],dec numero1[1],dec numero1[2],dec
numero1[3],dec numero1[4],dec numero1[5],dec numero1[6],dec numero1[7],dec numero1[8],34,13]
    pause 1000
    serout2 TX,188,["CH0=", DEC CH0, " CH1=",DEC CH1,26,13] 'ENVIO DE MENSAJE
    PAUSE 500
    return
MENSAJE_AVISO:
    IF A=0 THEN
        serout2 TX,188,["at+cmgs=",34,dec numero1[0],dec numero1[1],dec numero1[2],dec
numero1[3],dec numero1[4],dec numero1[5],dec numero1[6],dec numero1[7],dec numero1[8],34,13]
        pause 1000
        serout2 TX,188,["EXCESO DE CO !!! ","CH0=", DEC CH0, " CH1=",DEC CH1,26,13] 'ENVIO
DE MENSAJE
        PAUSE 500
        serout2 TX,188,["at+cmgs=",34,dec numero2[0],dec numero2[1],dec numero2[2],dec
numero2[3],dec numero2[4],dec numero2[5],dec numero2[6],dec numero2[7],dec numero2[8],34,13]
        pause 1000
        serout2 TX,188,["EXCESO DE CO !!! ","CH0=", DEC CH0, " CH1=",DEC CH1,26,13] 'ENVIO
DE MENSAJE
        PAUSE 500
        serout2 TX,188,["at+cmgs=",34,dec numero3[0],dec numero3[1],dec numero3[2],dec
numero3[3],dec numero3[4],dec numero3[5],dec numero3[6],dec numero3[7],dec numero3[8],34,13]
        pause 1000
        serout2 TX,188,["EXCESO DE CO !!! ","CH0=", DEC CH0, " CH1=",DEC CH1,26,13] 'ENVIO
DE MENSAJE
        PAUSE 500
        LCDOut $fe, 1
        LCDOut "Msg. Enviado!!!"
        PAUSE 500
        A=1
    ENDIF

```

FIG. 3.3 VISUALIZACION DE PARTE DEL PROGRAMA EN LA VENTANA DEL PBP

```

MicroCode Studio - PICBASIC PRO (medidor3.pbp)
File Edit View Project Help
New Open Save Cut Copy Paste Undo Redo Print
Compile Compile Program 16F877A Read Verify Erase Information
medidor3

*****
DEFINE OSC 4
'DEFINICIONES PARA LCD
DEFINE LCD_DREG PORTD
DEFINE LCD_DBIT 4
DEFINE LCD_RSREG PORTD
DEFINE LCD_RSBIT 3
DEFINE LCD_EREG PORTD
DEFINE LCD_EBIT 2
*****
'CARACTERISTICAS DEL ADC
DEFINE ADC_BITS 8
DEFINE ADC_CLOCK 3
DEFINE ADC_SAMPLEUS 50
TRISA = %11111111 ' Set PORTA to all input
ADCON1 = %00000000 ' Set PORTA analog
*****
'VARIABLES DEL ADC
ch0 VAR BYTE 'CO1
ch1 VAR BYTE 'CO2
canal0 VAR WORD
canal1 VAR WORD
*****
'PUERTOS PARA COMUNICACION CON MODEM GSM
TX VAR PORTB.6:RX VAR PORTB.7
*****
modem VAR BYTE [20]
*****
D0 VAR BYTE:D1 VAR BYTE:D2 VAR BYTE:D3 VAR BYTE
D4 VAR BYTE:D5 VAR BYTE:D6 VAR BYTE:D7 VAR BYTE

```

3.3.4. SOFTWARE DEL MEDIDOR DE CO PARA DISPOSITIVO MÓVIL

3.3.4.1 ESPECIFICACIONES DE LA APLICACIÓN PARA EL DISPOSITIVO MÓVIL

- La aplicación del dispositivo móvil debe tener las siguientes características:
- Comunicación Bluetooth con la tarjeta de adquisición de datos
- Medir y visualizar en pantalla el valor de CO
- Registrar una Grafica del valor de CO
- Guardar en un archivo jpg la grafica de los Valores Registrados

Teniendo en cuenta las especificaciones mencionadas se desarrolla la aplicación la cual tiene la pantalla que se muestra en la Figura 3.5 Donde se indican las funciones de cada componente.

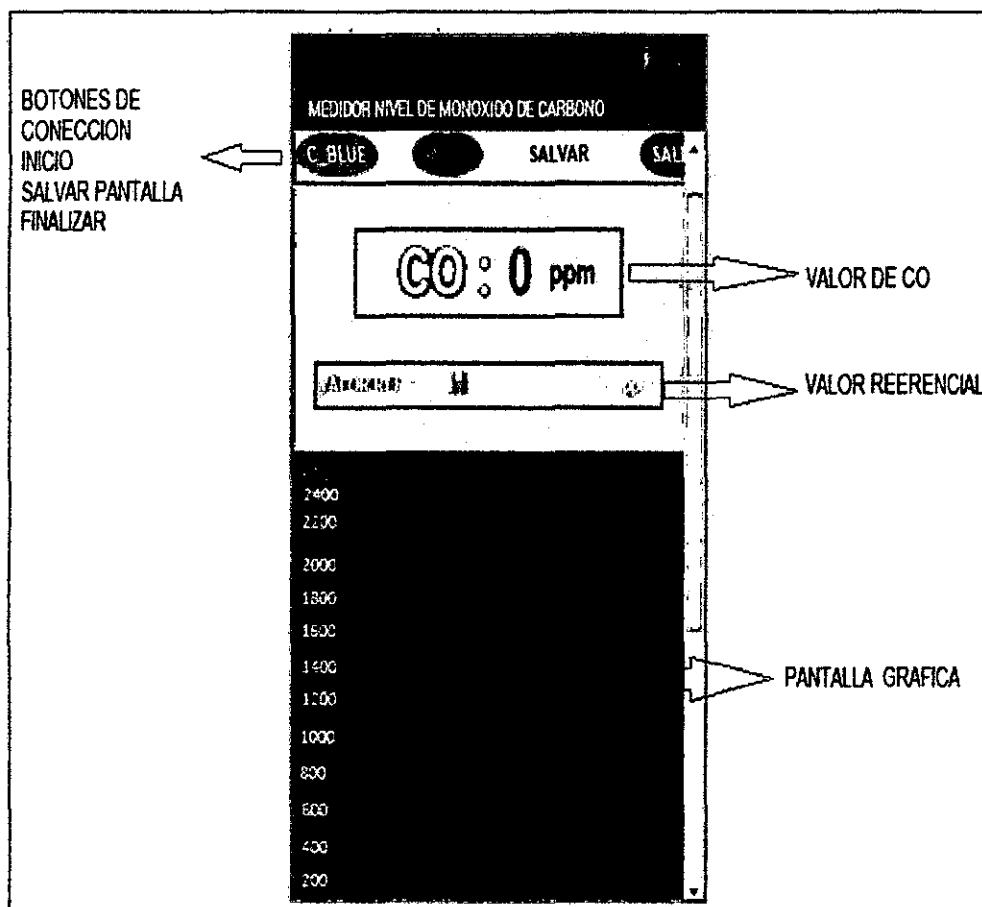


FIG. 3.4 PANTALLA PRINCIPAL DE LA APLICACIÓN EN DISPOSITIVO MÓVIL PARA MEDICIÓN DE CO

3.3.4.2 DESARROLLO DE APLICATIVO EN APPINVENTOR

Para el desarrollo de la aplicación se utilizó el appinventor2, el diseño de la pantalla se muestra en la Figura 3.5 y Figura 3.6, en ella se puede apreciar todos los componentes de la misma.

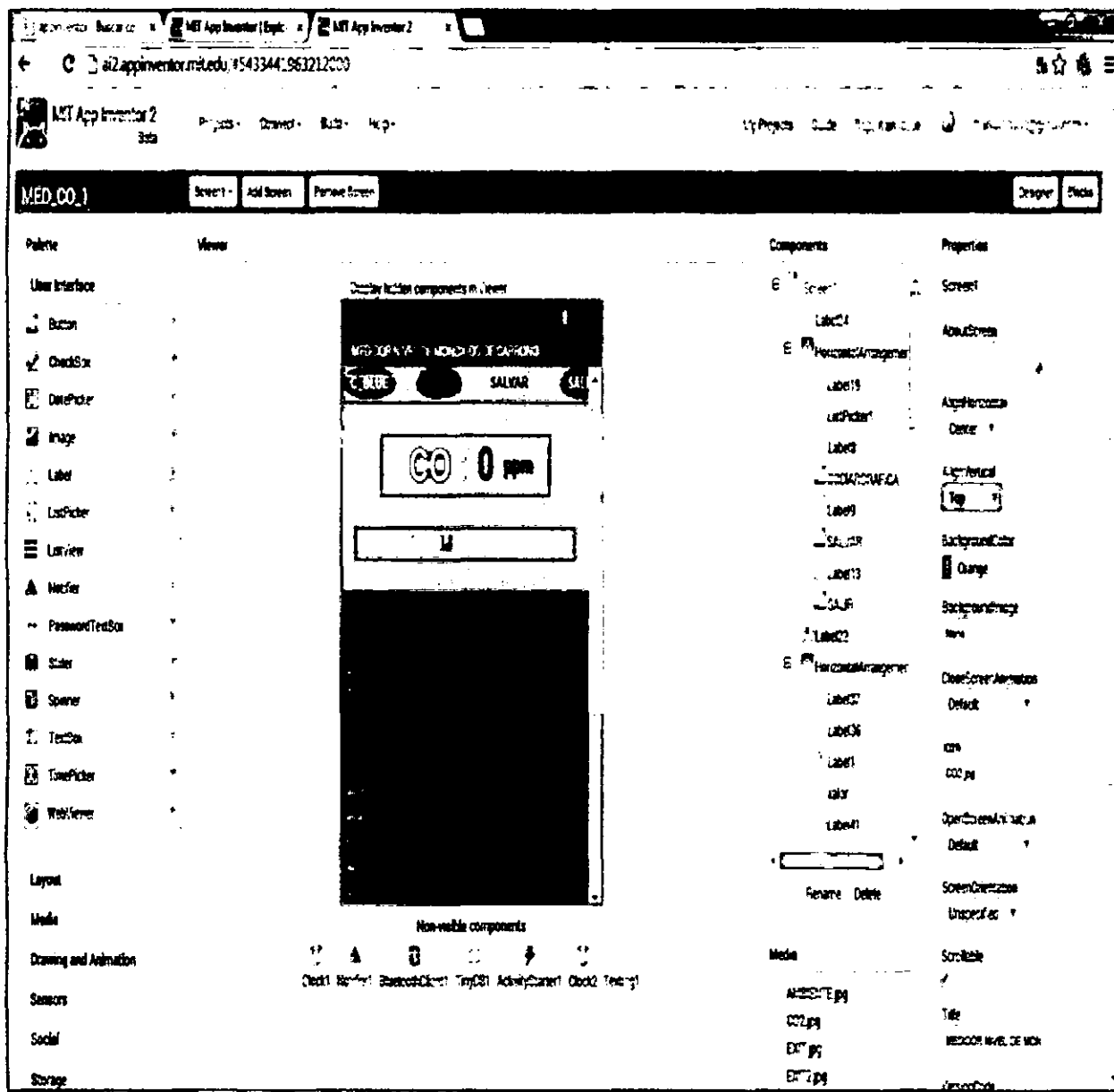


FIG. 3.5 PANTALLA DE DISEÑO DE APLICACIÓN EN APPINVENTOR2

3.3.4.2.1 COMPONENTES VISIBLES

Los componentes visibles son los que se muestran en la Figura 3.7 y esta compuesto por caja de texto , botones y etiquetas.

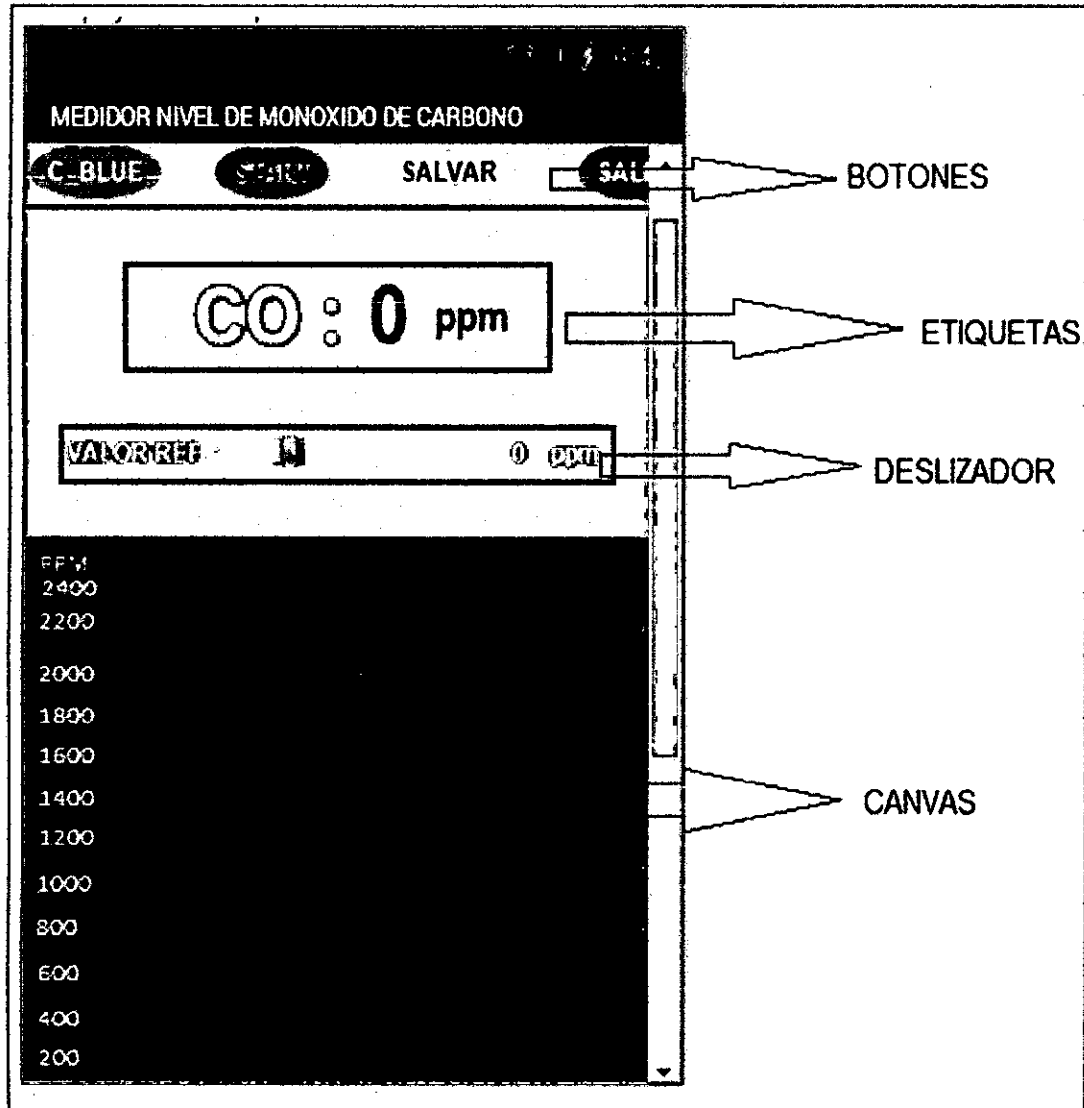


FIG. 3.6 COMPONENTES VISIBLES DE LA APLICACIÓN

3.3.4.2.2 COMPONENTES NO VISIBLES

Estos componentes no son visibles al momento de ejecutar la aplicación pero si son programables por el usuario según requerimientos, los componentes no visibles que se utiliza son los que se muestran en la Figura 3.8

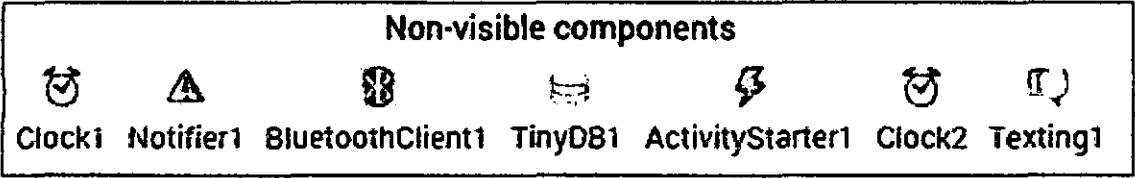


FIG. 3.7 COMPONENTES NO VISIBLES

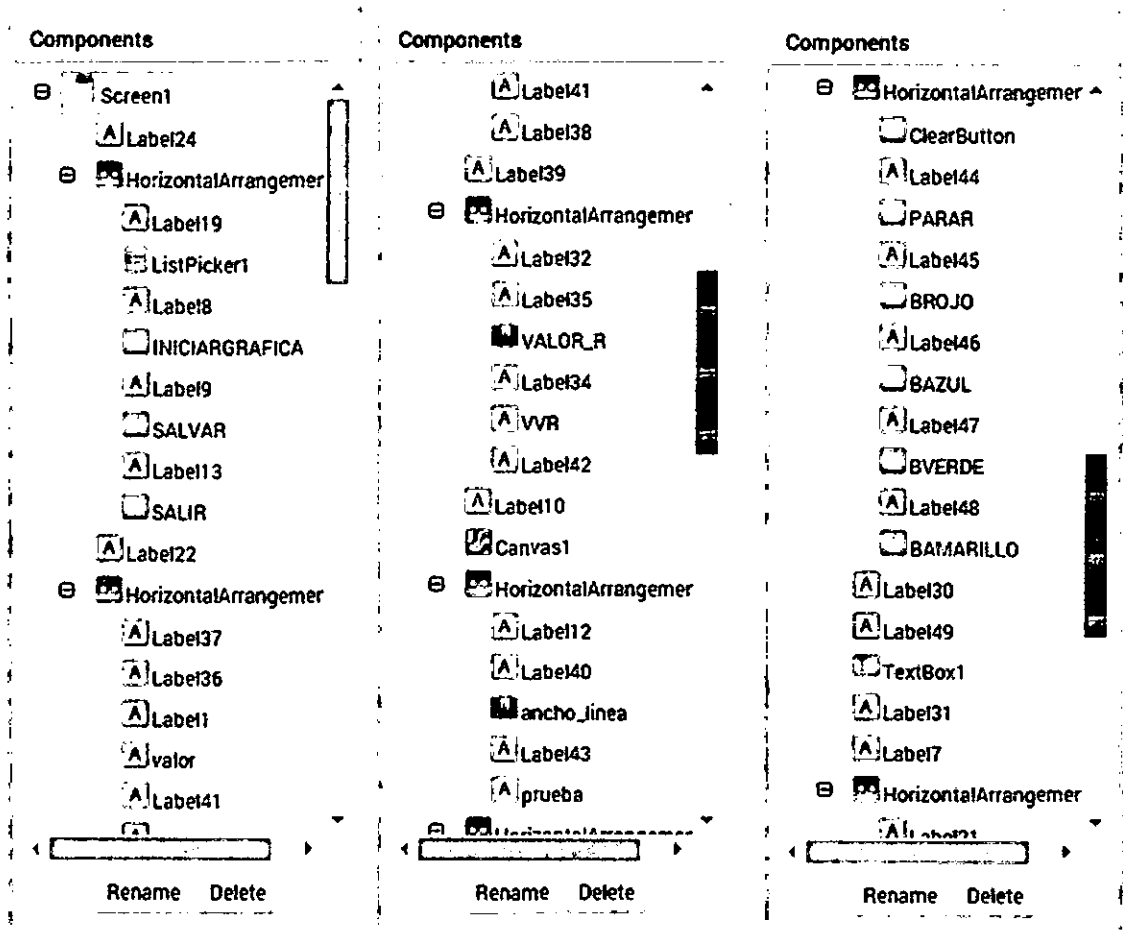


FIG. 3.8 NOMBRES DE LOS DIFERENTES COMPONENTES DE LA APLICACIÓN

3.3.4.3 DESCRIPCIÓN DEL PROGRAMA

3.3.4.3.1 VARIABLES

Las variables px1, px2, px3, px4, py1,py2, py3 y py4son utilizadas para graficar línea por línea, uniendo dos puntos (px1, py1; px2,py2; px3, py3; px4, py4). La variable k tomará el valor de 1 si el dispositivo móvil recibe la cabecera de la cadena (A1). La variable VR almacenará el valor de referencia. La variable t=4, es para indicar el inicio de la gráfica en la pantalla (eje x). La variable i es utilizada para la creación del archivo con extensión .jpg. La variable SMS es utilizada para activar el envío del SMS.

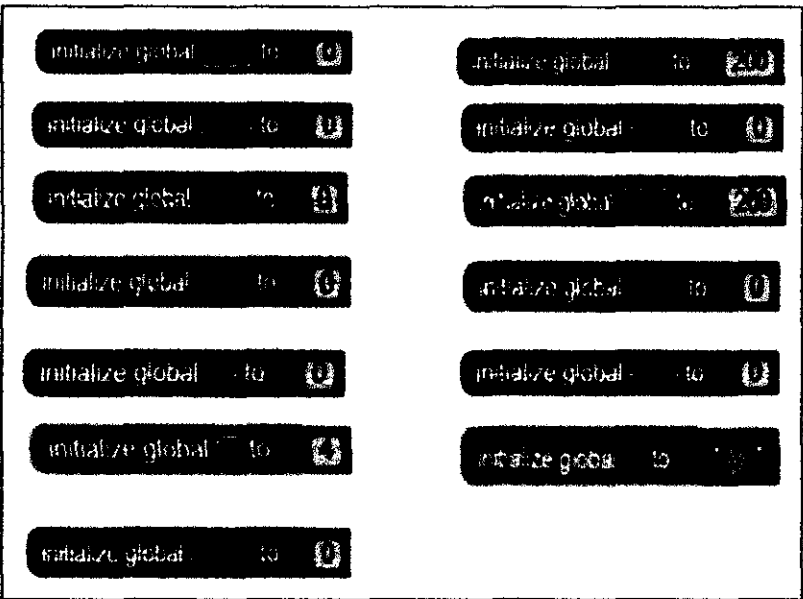


FIG 3.9 DISEÑO DE DECLARACIÓN DE VARIABLES DEL EDITOR DE BLOQUES DE APP INVENTOR2

3.3.4.3.2 BOTONES

BOTÓN CONECTAR (C_BLUE):

Al hacer click en el botón CONECTAR, establecemos comunicación con el módulo Bluetooth HC-06, si está en ON el Bluetooth del celular, hace un registro de los Bluetooth sincronizados, seleccionamos el que vamos a utilizar y luego se muestra un mensaje: “BLUETOOTH HABILITADO”. Finalmente se conectará al Bluetooth.

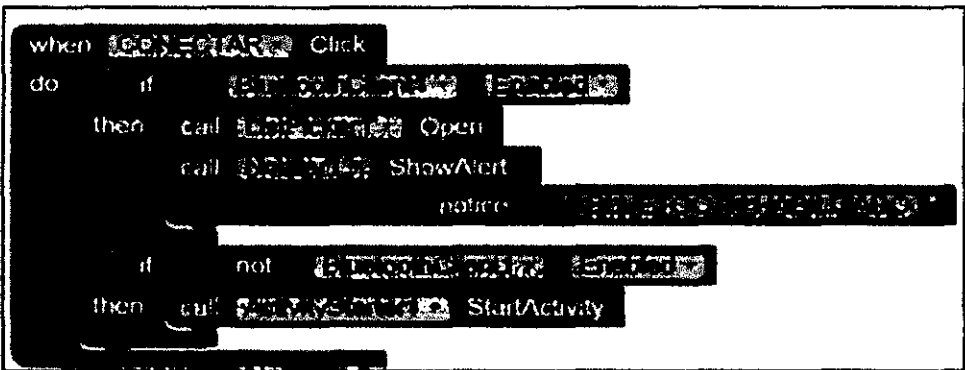


FIG. 3.10 DISEÑO DEL BOTÓN CONECTAR DEL EDITOR DE BLOQUES DE APP INVENTOR2

BOTÓN LISTPICKER1- BEFOREPICKING

Al hacer clic en el botón ListPicker1, el Bluetooth busca al dispositivo sincronizado, si el módulo Bluetooth del Celular después de buscar un registro, y no se dan resultados, se asume que el módulo Bluetooth está y se genera una notificación: “Por Favor Encender Bluetooth”.

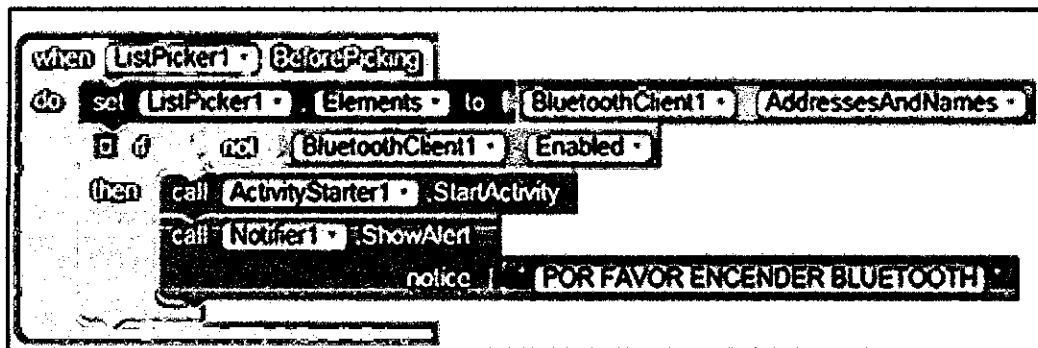


FIG. 3.11 LISTPICKER (BEFOREPICKING)

BOTÓN LISTPICKER1- AFTERPINCKING

Al hacer click en el Botón ListPicker1 – AfterPicking se nos muestra una lista de los Bluetooth Sincronizados. Se conecta con el Módulo Bluetooth HC-06 ingresamos su número de MAC y pintamos el indicador de conección de color Azul.

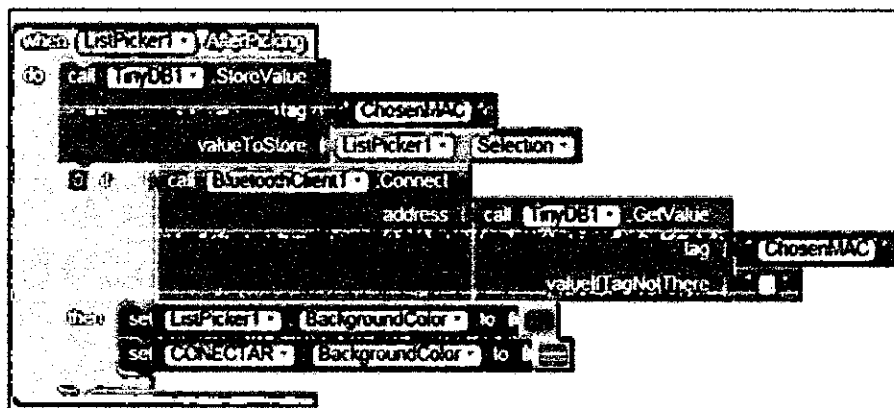


FIG. 3.12 DISEÑO DEL BOTÓN LISTPICKER1

BOTÓN SALIR

Al hacer click en el botón SALIR, cortamos comunicación con el Módulo Bluetooth HC -06 y cerramos la aplicación.

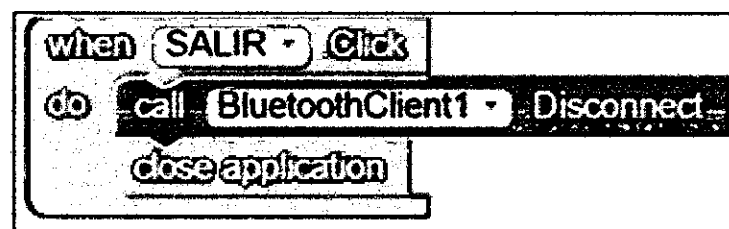


FIG 3.13 DISEÑO DEL BOTÓN SALIR

BOTÓN PARAR

Al hacer click en el botón PARAR, desactivamos el temporizador (Clock1)



FIG. 3.14 DISEÑO DEL BOTÓN PARAR

BOTÓN SALVAR

Al hacer click en el botón SALVAR, utilizamos el evento notificación para asignar el nombre de la imagen a guardar.

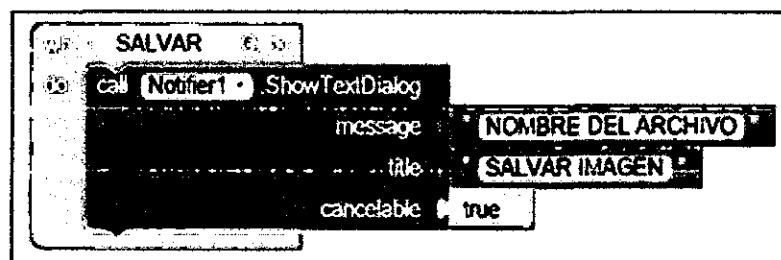


FIG. 3.15 DISEÑO DEL BOTÓN SALVAR

BOTÓN REDBUTTON

Al hacer click en el botón RedButton activamos el temporizador (Clock 1).

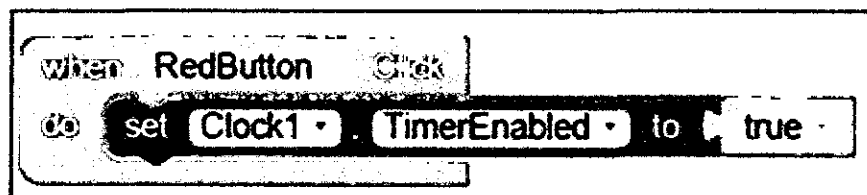


FIG. 3.16 DISEÑO DEL BOTÓN REDBUTTON

BOTÓN CLEARBUTTON

Al hacer click en el botón ClearButton, asignamos inicio de la gráfica en eje Horizontal ($t=4$), limpiamos la herramienta de animación de la gráfica (canvas) y desactivamos el temporizador (Clock1).

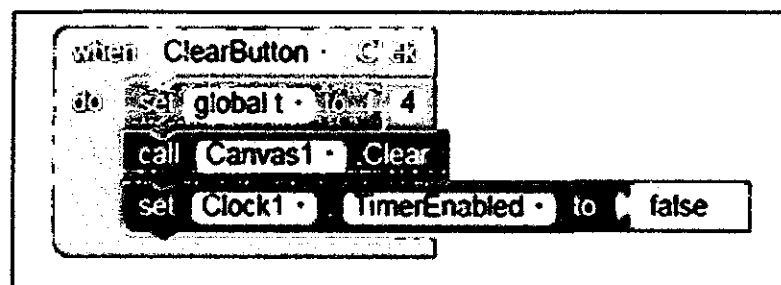


FIG 3.17 DISEÑO DEL BOTÓN CLEARBUTTON

3.3.4.3.3 DESLIZADOR

DESLIZADOR SILDER1

Al variar el deslizador Silder1, éste asigna valores numéricos entre 2-200 para varia Línea de referencia.

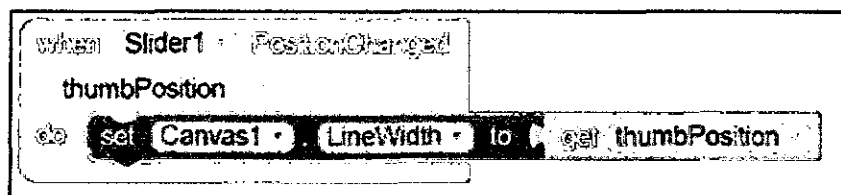


FIG. 3.18 VARIACIÓN DE LÍNEA DE REFERENCIA

3.3.4.3.4 TEMPORIZADOR

TEMPORIZADOR CLOCK1

Se recibe la cadena que envía el Módulo Bluetooth HC-06 de 12 bytes (A1025B1140F13), si el dispositivo móvil recibe la cabecera de la Cadena (A1) entonces k tomará el valor de 1. Luego se compara, si es k es igual a 1 entonces, se extrae un segmento de la cadena desde el byte 2 hasta el byte 4 (1025) obteniendo 1025, a continuación le restamos 1000, obteniendo como resultando el valor del sensor de Temperatura (25), este clock1 guiándose de los 12 bytes que recibe, se obtiene los valores de las variables (temperatura, nivel).

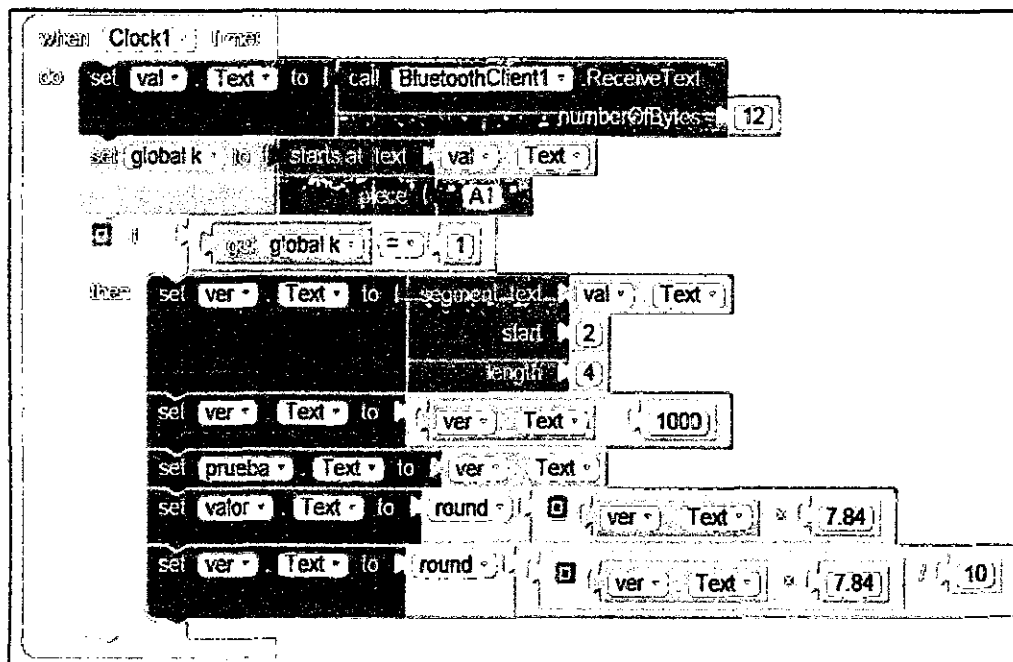


FIG. 3.19 DISEÑO DEL CLOCK1 DEL EDITOR DE BLOQUES DE APPINVENTOR

TEMPORIZADOR CLOCK2

Este temporizador grafica las variable del CO en el Canvas. Px y Py son las coordenadas o puntos que al unirse forman una recta. El ancho de la pantalla donde se grafica es de los 320 píxeles que al momento de unirse forman una recta de color ROJO.

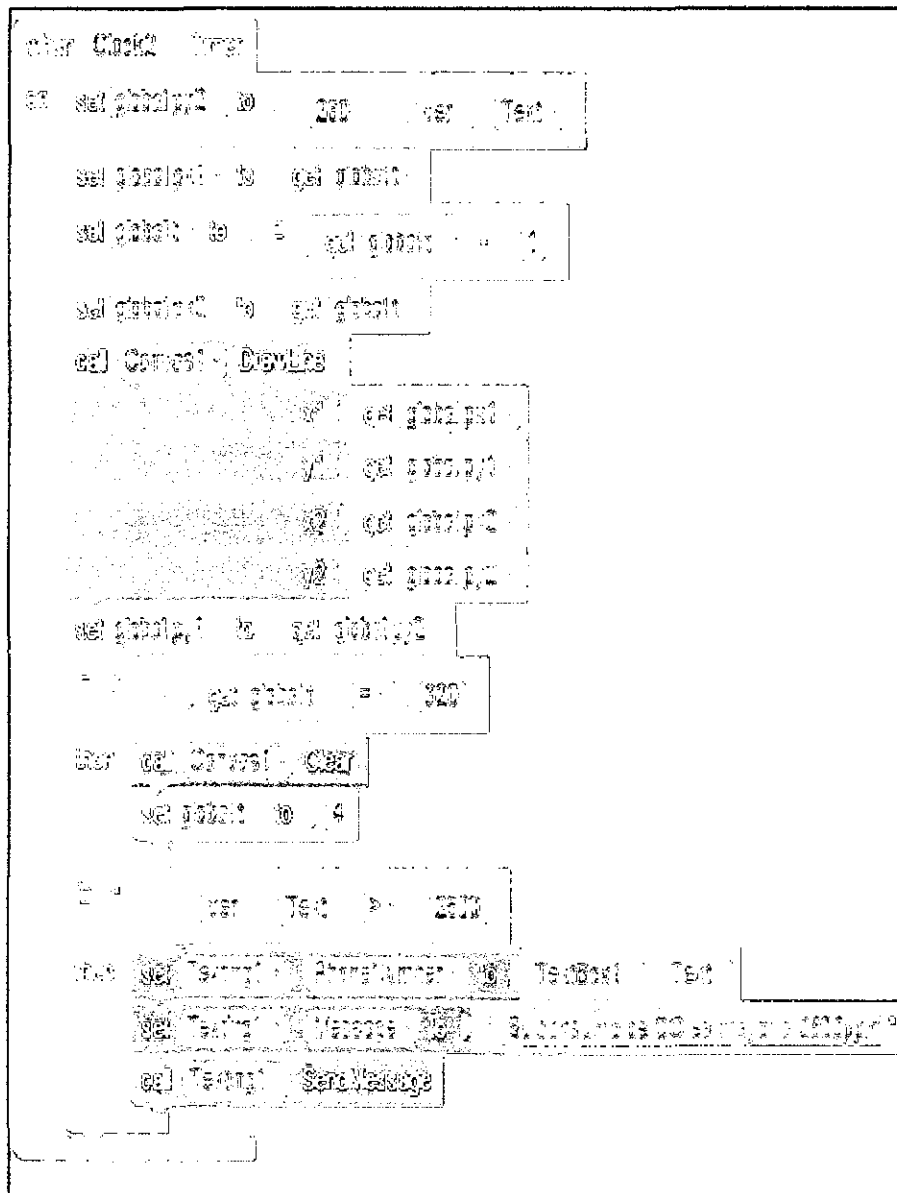


FIG. 3.20 DISEÑO DEL CLOCK2

3.3.4.3.5 NOTIFICACIÓN

NOTIFICACIÓN NOTIFIER1

Después de introducido el nombre se guardará la imagen (canvas) con extensiones .jpg, y se visualizará una notificación indicando que la imagen ha sido salvada.

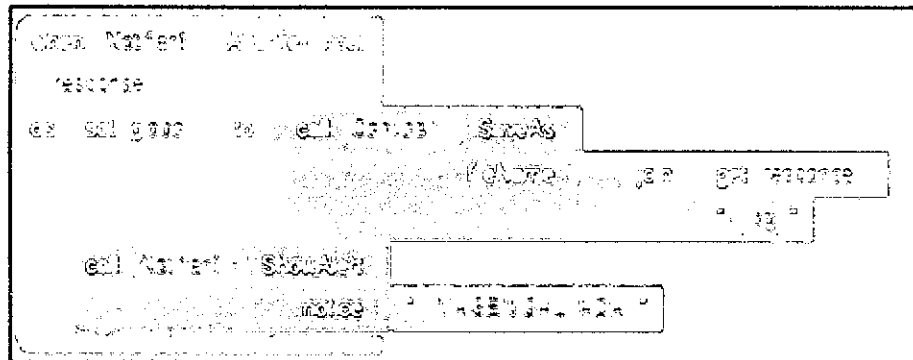


FIG. 3.21 DISEÑO DE NOTIFICACIÓN

TEXTING1

Se hace una comparación entre los rangos permitidos (ver1 y ver2) y los obtenidos, si los datos obtenidos pasan del rango permitido se enviará un mensaje de texto en forma de alarma al número del celular ingresado.

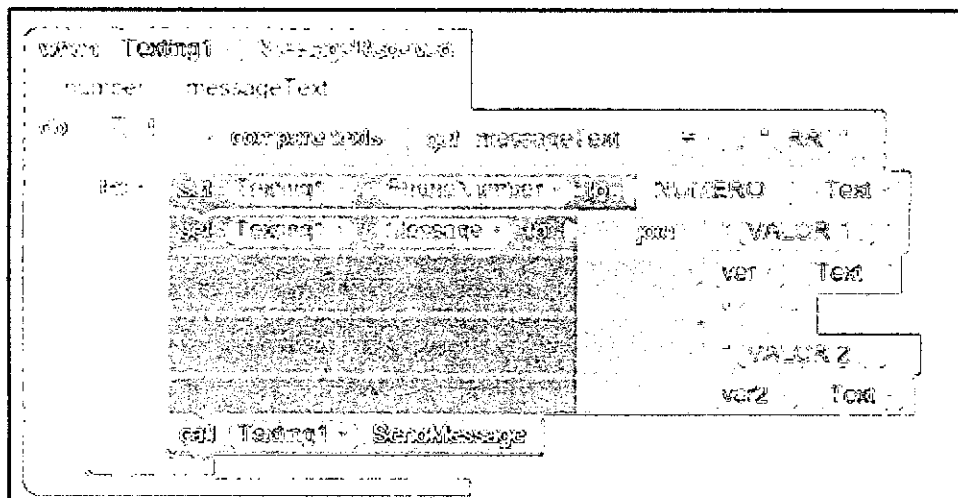


FIG. 3.22 DISEÑO DE TEXTING1 (ENVIAR MENSAJE DE TEXTO)

CAPITULO IV

4.0 ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

4.1 PRUEBAS Y RESULTADOS AL PROTOTIPO

Se llevaron a cabo las pruebas realizadas con el prototipo para verificar el correcto funcionamiento del sistema y comprobar su uso de manera práctica.

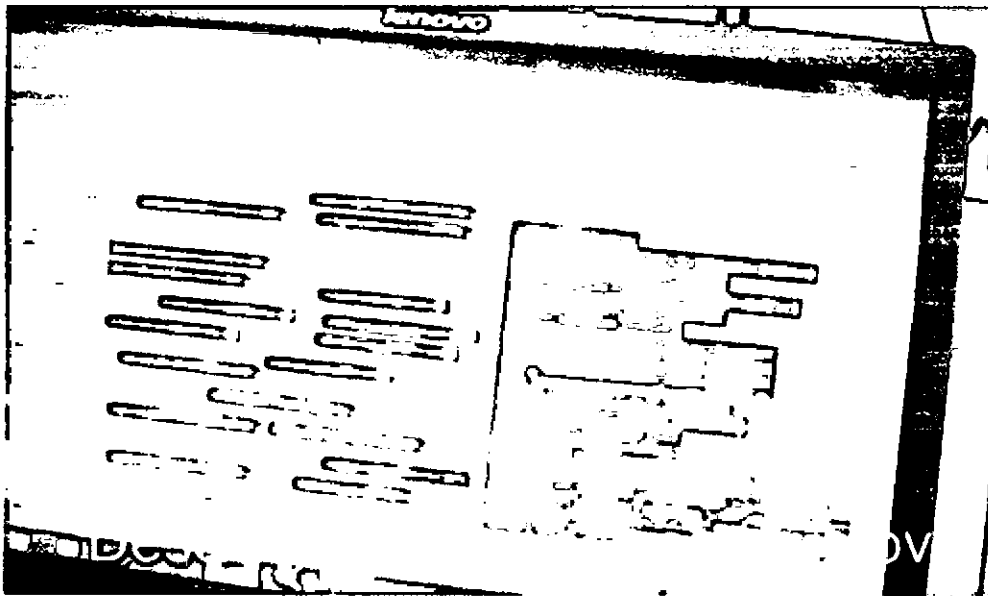


FIG. 4.0 PROGRAMA VISUALIZADO EN APP INVENTOR



FIG. 4.1 APLICACIÓN VISUALIZADA EN UNA TABLET



FIG. 4.2 MEDICIÓN DEL HUMO DE CIGARRO



FIG. 4.3 MEDICIÓN VISUALIZADA EN UN CELULAR

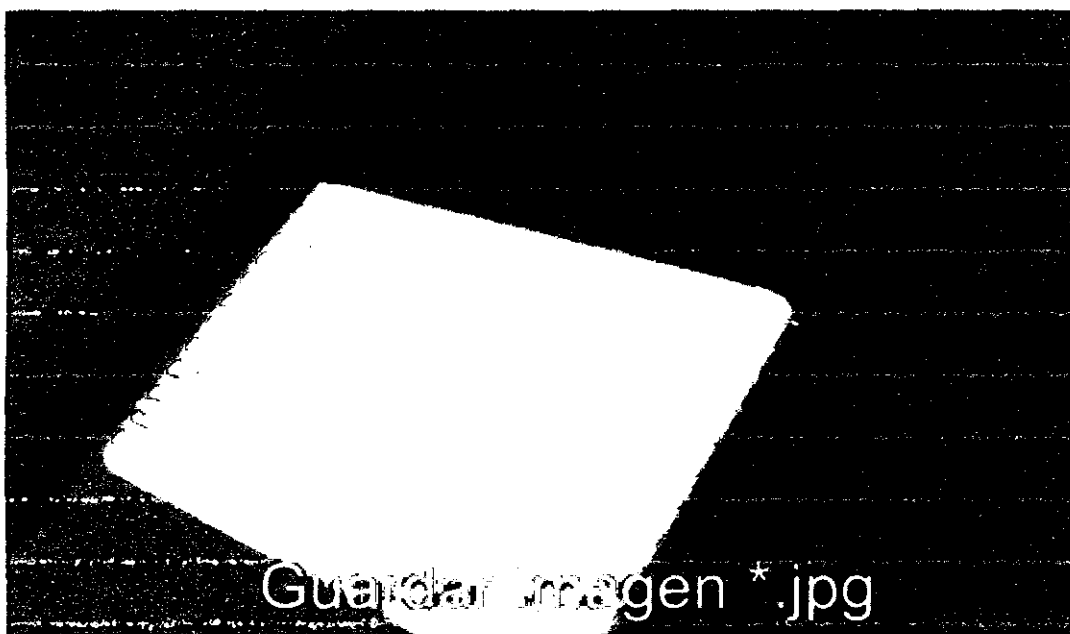


FIG. 4.4 GRAFICA ADQUIRIDA DE LOS VALORES DEL SENSOR

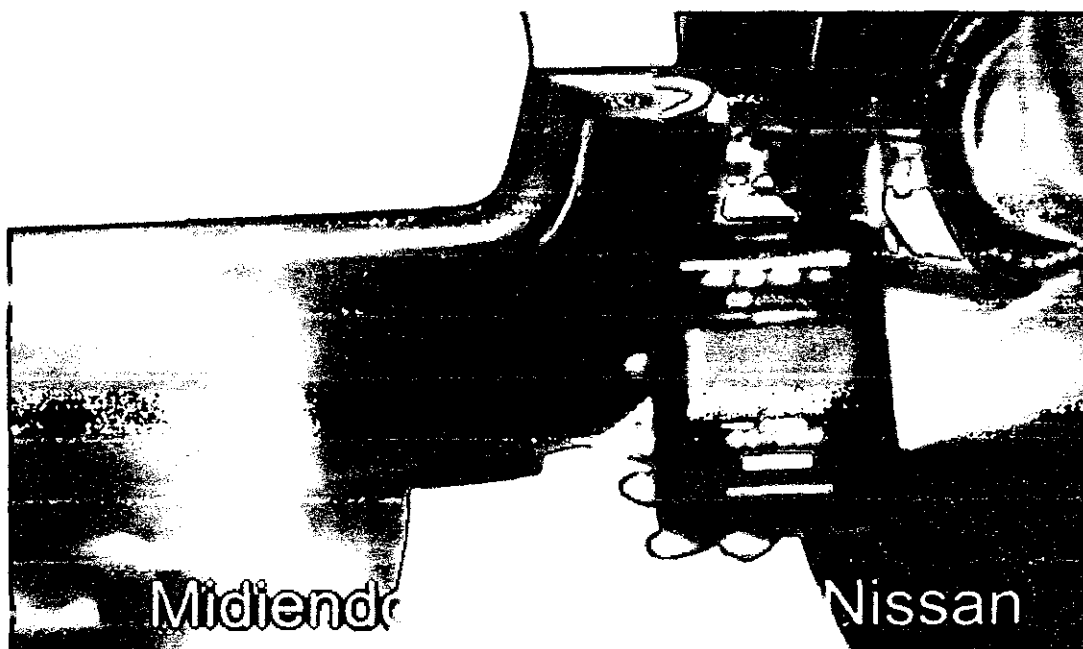


FIG. 4.5 MEDIDA DE CO EN UN AUTO NISSAN



FIG. 4.6 MEDICIÓN DE CO EN UN AUTO SUZUKI



FIG. 4.7 VISUALIZACION DE LA MEDIDA DEL CO EN UNA TABLET

4.2 CONTRASTACIÓN DE LA HIPÓTESIS

Usando las tecnologías existentes de sensores, de monóxido de carbono microcontroladores PIC 16F877A, y demás circuitos integrados así como seleccionando el software PIC BASIC PRO y APP Inventor se pudo diseñar un sistema de monitoreo de contaminación ambiental, producido por los vehículos de transporte público.

Se llevaron a cabo las pruebas necesarias con diferentes tipos de gases y se pudo graficar y apreciar los valores emitidos de estos gases, por ello decimos que con estos dispositivos económicos y sencillos se pudo construir un sistema de uso eficiente a la hora de evaluar la emisión de gases emitidos por los motores de los vehículos.

CAPITULO V

5.0 CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- Se diseñó un sistema de monitoreo de la contaminación ambiental, producida por los vehículos de transporte público.
- Diseñamos un sistema de detección y/o medición de emisión de monóxido de carbono y otros gases que contribuyen a la contaminación ambiental.
- El sistema de comunicación GSM, GPRS se utilizó dentro de la aplicación y se comprendió la importancia y factibilidad de su uso.
- Se diseñó el software para dispositivo móvil en android y poder utilizarlo en cualquier móvil con ese sistema operativo, de fácil uso y accesible a todos los usuarios.
- Se diseñó el software para PC de escritorio utilizado para la programación grabada en el Pic y un emulador de android para cargar la aplicación en la Pc.

5.2 RECOMENDACIONES

- El diseño y construcción, hardware y software deben ir de la mano, de preferencia se recomienda realizar el diseño completo del hardware para luego tener una idea más clara de cómo desarrollar un programa (software) óptimo.
- Realizar la correcta elección del sensor ya que algunos presentan deficiencias ante ambientes con alto grado de suciedad, temperatura, etc.
- Contar con los diferentes programas desarrolladores de circuitos completos y actualizados ya que las versiones de prueba retrasan el desarrollo de pruebas del sistema.
- En el mercado local (Tiendas electrónicas) algunos componentes no se hayan, por esta razón se debe realizar búsquedas y compras de estos componentes en la red (Tiendas online) con tiempo, ya que se debe considerar el tiempo de demora en el envío y a ello se suma el retraso del proyecto.
- Se recomienda realizar varias pruebas de medición para un mejor entendimiento del sistema y un buen monitoreo óptimo.
- Investigar las diferentes tendencias tecnológicas de actualidad, aplicables al campo de la electrónica, para mejorar el desarrollo e implementación de proyectos y facilitar la comprensión de los diferentes procesos que implica desarrollar un sistema.
- Terminado el proyecto final realizar varias pruebas para comprobar el perfecto funcionamiento de la aplicación desarrollada.

BIBLIOGRAFIA

- [1] <http://www.parallax.com/education>, en línea, [Citado el 15 de julio del 2014]
- [2] <http://doru.info/projects/hdl/pavr/index.html>, en línea, [Citado el 18 de junio del 2014]
- [3] Microcontroladores Pic, programación en basic. Carlos A. Reyes, 2da edic. pág. 18
- [4] <http://www.mikroe.com/chapters/view/84/libro>, en línea, [Citado el 25 de julio del 2014]
- [5] G. COLOURIS, J. DOLLIMORE, TIM, KINDBERG, SISTEMAS DISTRIBUIDOS, Conceptos y Diseño, Tercera edición, Editorial Addison Wesley 2001, pág. 79
- [6] Ferran Casadevall y Oriol Sallent. "Sistema GPRS". UPC Barcelona. Mayo 2002. Pág. 17
- [7] Ferran Casadevall y Oriol Sallent. "Sistema GPRS". UPC. Barcelona. Mayo 2002. Pág. 21
- [8] <http://www.lcd.usb.ve/figueira/cursos/redes2/EXPO-em01/Bluetooth/bluetooth.htm>, en línea, [citado el 16 de noviembre del 2014]
- [9] Android Developers. Design. [En línea]. <http://developer.android.com/guide/index.html>. [Citado el 16 de agosto del 2014]
- [10] www.amena.com/presentacion/particulares/telefonos/gprs/, en línea, [Consultado el 16 de noviembre del 2014]
- [11] www.wmlclub.com/articulos/fundamentosgsm.htm, en línea, [Consultado el 17 de octubre del 2014]
- [12] Universidad Carlos III de Madrid. Programación en dispositivos móviles portables. Arquitectura Android, en línea <https://sites.google.com/site/swcuc3m/home/android/generalidades/2-2-arquitectura-de-android>. [Citado el 02 de noviembre del 2014]
- [13] Definición de móvil, en línea. <http://definicion.de/movil/>. [Citado el 21 de Noviembre del 2015]
- [14] Master Magazine. Artículo 160, en línea, [Citado el 26 de octubre del 2014]
<http://www.mastermagazine.info/articulo/610.php>. [Citado en 20 de Diciembre del 2014]
- [15] GSMspain. Glosario. Smartphone, en línea,
www.gsmspain.com/glosario/?palabra=SMARTPHONE. [Citado en 21 de Junio del 2014]
- [16] Real Academia Española. Sistema Operativo, en línea,
http://buscon.rae.es/draeI/SrvltObtenerHtml?origen=RAE&LEMA=sistema&SUPIN=D=0&CAREXT=10000&NEDIC=No#sistema_operativo. [Citado el 21 de agosto del 2014]

- [17] Universidad de Granada. Departamento de Ciencias de la Computación e Inteligencia Artificial. Desarrollo de aplicaciones para dispositivos móviles, en línea http://leo.ugr.es/J2ME/INTRO/intro_9.htm [citado en 02 de Octubre del 2012]
- [18] OSL. Oficina de Software Libre. Universidad de las palmas de Gran Canaria, creación de aplicaciones móviles para android, en línea, <http://osl.ulpgc.es/cursos/android>. [Citado el 02 de Setiembre de 2014]
- [19] Androidsis. Android arrasa en ventas, en línea, <http://www.androidsis.com/android-arrasa-en-ventas/>. [Citado en 06 de Junio del 2014]
- [20] ADSL ZONE. LinuxZone. Kernel, en línea, <http://www.linuxzone.es/que-es-el-kernel/>. [Citado el 23 de Noviembre del 2014]
- [21] Raw thoughts, Technology, Experiences. Kernel, en línea, <http://ranganaths.wordpress.com/2010/07/03/android-kernel-and-linux-kernel/>. [Citado el 26 de agosto del 2014]
- [22] Android Authority. Open Core, en línea, <http://www.androidauthority.com/packetvideo-release-opencore-20-for-android-2099/>. [Citado en 12 de Setiembre del 2014]
- [23] GARGENTA, Marko. Learning Android. Runtime. Estados Unidos. Andy Oram, Brian Jepson, 2011. 239p.



PIC16F87XA

28/40/44-Pin Enhanced Flash Microcontrollers

Devices Included In this Data Sheet:

- PIC16F873A
- PIC16F874A
- PIC16F876A
- PIC16F877A

High-Performance RISC CPU:

- Only 35 single-word instructions to learn
- All single-cycle instructions except for program branches, which are two-cycle
- Operating speed: DC – 20 MHz clock input
DC – 200 ns instruction cycle
- Up to 8K x 14 words of Flash Program Memory,
Up to 368 x 8 bytes of Data Memory (RAM),
Up to 256 x 8 bytes of EEPROM Data Memory
- Pinout compatible to other 28-pin or 40/44-pin
PIC16CXXX and PIC16FXXX microcontrollers

Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler,
can be incremented during Sleep via external
crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period
register, prescaler and postscaler
- Two Capture, Compare, PWM modules
 - Capture is 16-bit, max. resolution is 12.5 ns
 - Compare is 16-bit, max. resolution is 200 ns
 - PWM max. resolution is 10-bit
- Synchronous Serial Port (SSP) with SPI™
(Master mode) and I²C™ (Master/Slave)
- Universal Synchronous Asynchronous Receiver
Transmitter (USART/SCI) with 9-bit address
detection
- Parallel Slave Port (PSP) – 8 bits wide with
external RD, WR and CS controls (40/44-pin only)
- Brown-out detection circuitry for
Brown-out Reset (BOR)

Analog Features:

- 10-bit, up to 8-channel Analog-to-Digital
Converter (A/D)
- Brown-out Reset (BOR)
- Analog Comparator module with:
 - Two analog comparators
 - Programmable on-chip voltage reference
(VREF) module
 - Programmable input multiplexing from device
inputs and internal voltage reference
 - Comparator outputs are externally accessible

Special Microcontroller Features:

- 100,000 erase/write cycle Enhanced Flash
program memory typical
- 1,000,000 erase/write cycle Data EEPROM
memory typical
- Data EEPROM Retention > 40 years
- Self-reprogrammable under software control
- In-Circuit Serial Programming™ (ICSP™)
via two pins
- Single-supply 5V In-Circuit Serial Programming
- Watchdog Timer (WDT) with its own on-chip RC
oscillator for reliable operation
- Programmable code protection
- Power saving Sleep mode
- Selectable oscillator options
- In-Circuit Debug (ICD) via two pins

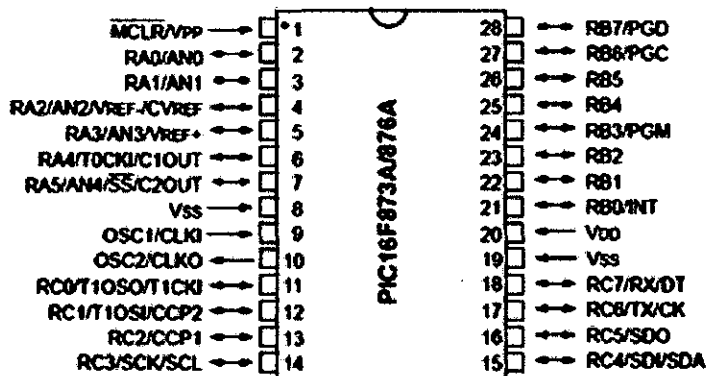
CMOS Technology:

- Low-power, high-speed Flash/EEPROM
technology
- Fully static design
- Wide operating voltage range (2.0V to 5.5V)
- Commercial and Industrial temperature ranges
- Low-power consumption

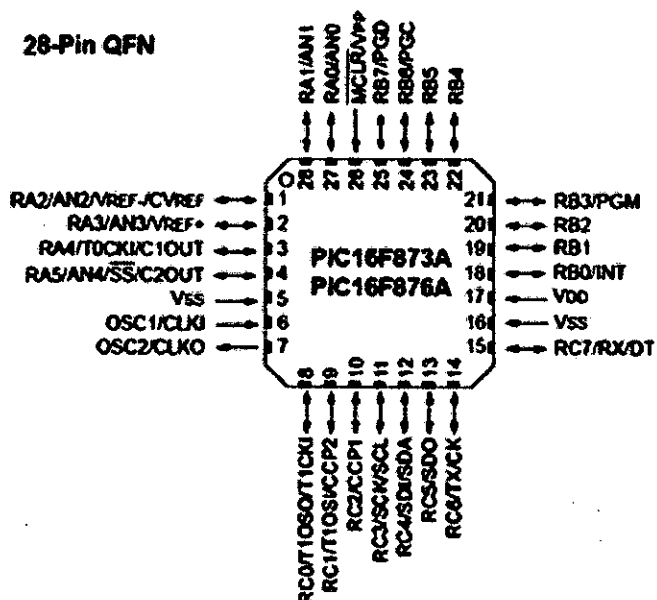
Device	Program Memory		Data SRAM (Bytes)	EEPROM (Bytes)	I/O	10-bit A/D (ch)	CCP (PWM)	MSSP		USART	Timers 8/16-bit	Comparators
	Bytes	# Single Word Instructions						SPI	Master I²C			
PIC16F873A	7.2K	4096	192	128	22	5	2	Yes	Yes	Yes	2/1	2
PIC16F874A	7.2K	4096	192	128	33	8	2	Yes	Yes	Yes	2/1	2
PIC16F876A	14.3K	8192	368	256	22	5	2	Yes	Yes	Yes	2/1	2
PIC16F877A	14.3K	8192	368	256	33	8	2	Yes	Yes	Yes	2/1	2

Pin Diagrams

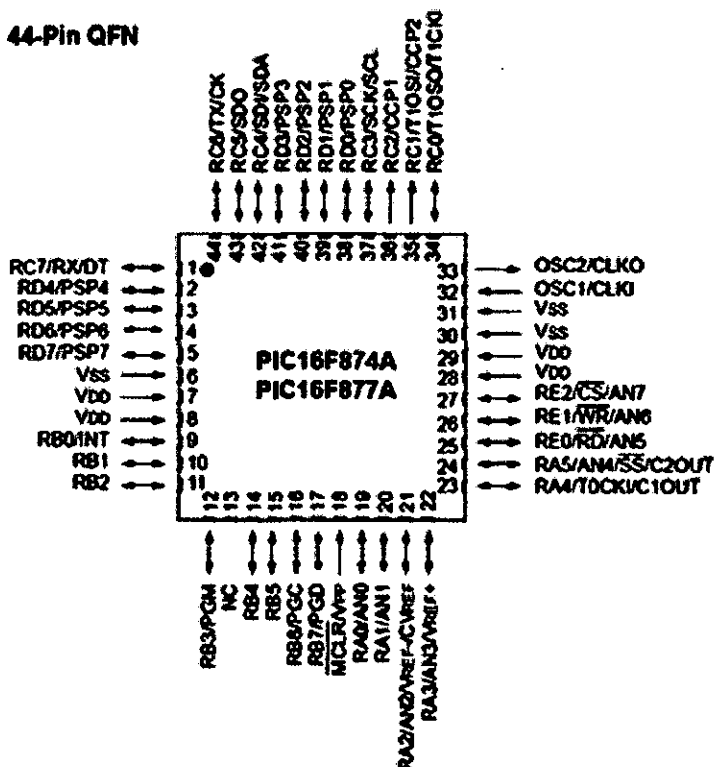
28-Pin PDIP, SOIC, SSOP



28-Pin QFN

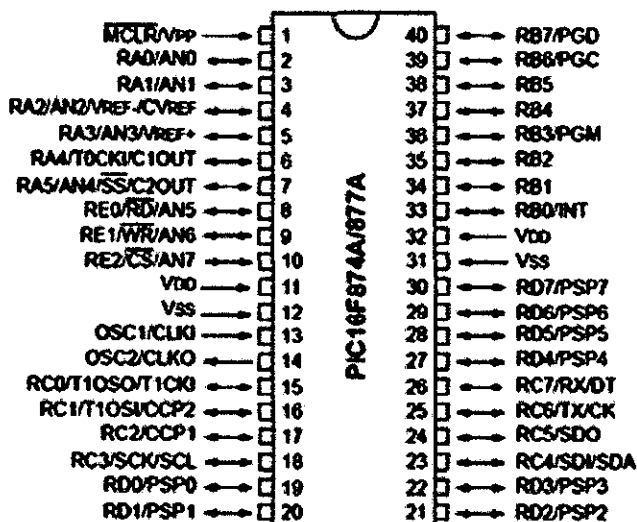


44-Pin QFN

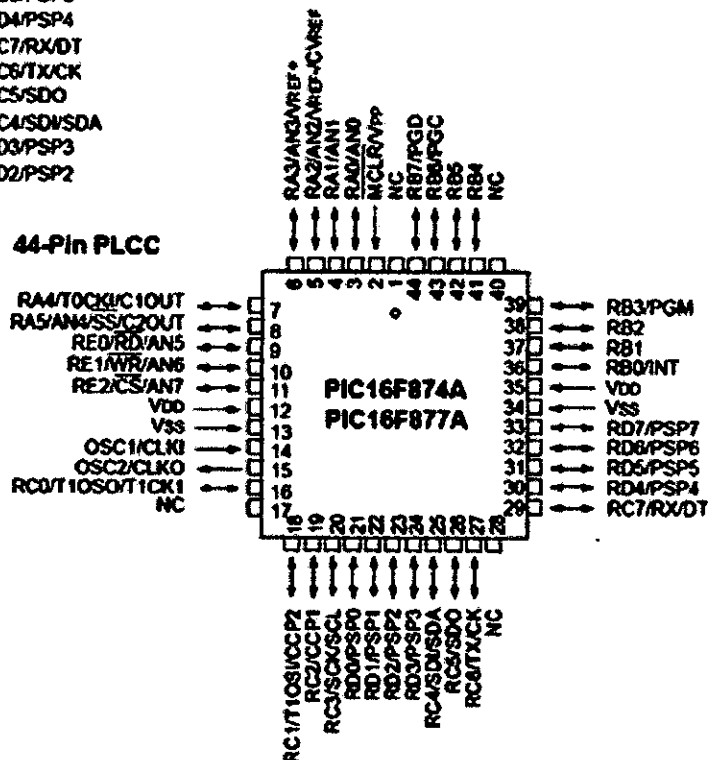


Pin Diagrams (Continued)

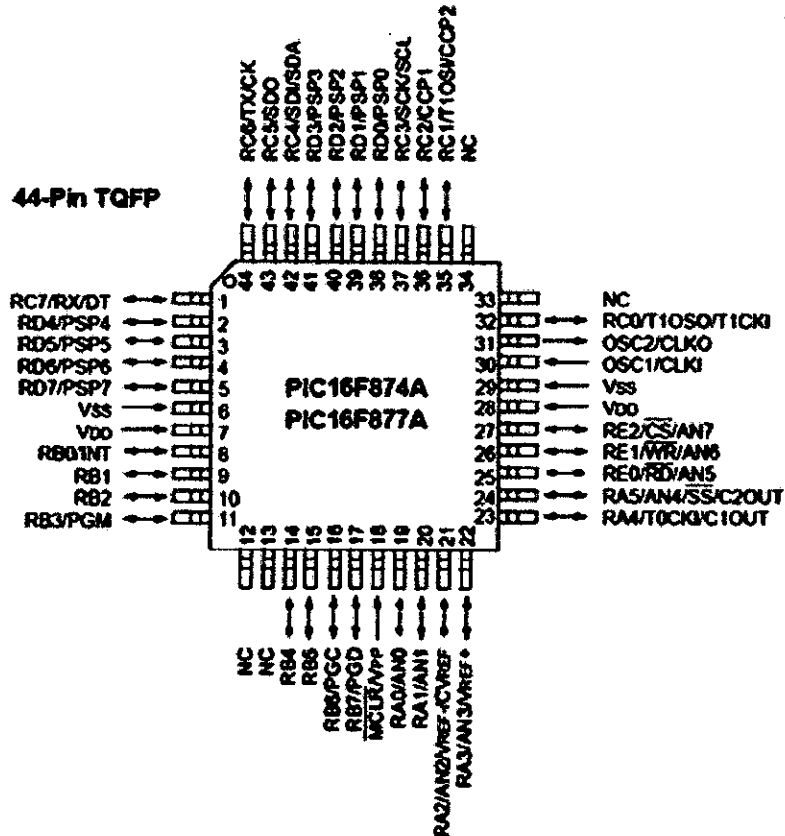
40-Pin PDIP



44-Pin PLCC



44-Pin TQFP



1.0 DEVICE OVERVIEW

This document contains device specific information about the following devices:

- PIC16F873A
- PIC16F874A
- PIC16F876A
- PIC16F877A

PIC16F873A/876A devices are available only in 28-pin packages, while PIC16F874A/877A devices are available in 40-pin and 44-pin packages. All devices in the PIC16F87XA family share common architecture with the following differences:

- The PIC16F873A and PIC16F874A have one-half of the total on-chip memory of the PIC16F876A and PIC16F877A
- The 28-pin devices have three I/O ports, while the 40/44-pin devices have five
- The 28-pin devices have fourteen interrupts, while the 40/44-pin devices have fifteen
- The 28-pin devices have five A/D input channels, while the 40/44-pin devices have eight
- The Parallel Slave Port is implemented only on the 40/44-pin devices

The available features are summarized in Table 1-1. Block diagrams of the PIC16F873A/876A and PIC16F874A/877A devices are provided in Figure 1-1 and Figure 1-2, respectively. The pinouts for these device families are listed in Table 1-2 and Table 1-3.

Additional information may be found in the PICmicro® Mid-Range Reference Manual (DS33023), which may be obtained from your local Microchip Sales Representative or downloaded from the Microchip web site. The Reference Manual should be considered a complementary document to this data sheet and is highly recommended reading for a better understanding of the device architecture and operation of the peripheral modules.

TABLE 1-1: PIC16F87XA DEVICE FEATURES

Key Features	PIC16F873A	PIC16F874A	PIC16F876A	PIC16F877A
Operating Frequency	DC – 20 MHz	DC – 20 MHz	DC – 20 MHz	DC – 20 MHz
Resets (and Delays)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)
Flash Program Memory (14-bit words)	4K	4K	8K	8K
Data Memory (bytes)	192	192	368	368
EEPROM Data Memory (bytes)	128	128	256	256
Interrupts	14	15	14	15
I/O Ports	Ports A, B, C	Ports A, B, C, D, E	Ports A, B, C	Ports A, B, C, D, E
Timers	3	3	3	3
Capture/Compare/PWM modules	2	2	2	2
Serial Communications	MSSP, USART	MSSP, USART	MSSP, USART	MSSP, USART
Parallel Communications	—	PSP	—	PSP
10-bit Analog-to-Digital Module	5 input channels	8 input channels	5 input channels	8 input channels
Analog Comparators	2	2	2	2
Instruction Set	35 Instructions	35 Instructions	35 Instructions	35 Instructions
Packages	28-pin PDIP 28-pin SOIC 28-pin SSOP 28-pin QFN	40-pin PDIP 44-pin PLCC 44-pin TQFP 44-pin QFN	28-pin PDIP 28-pin SOIC 28-pin SSOP 28-pin QFN	40-pin PDIP 44-pin PLCC 44-pin TQFP 44-pin QFN

FIGURE 1-1: PIC16F873A/876A BLOCK DIAGRAM

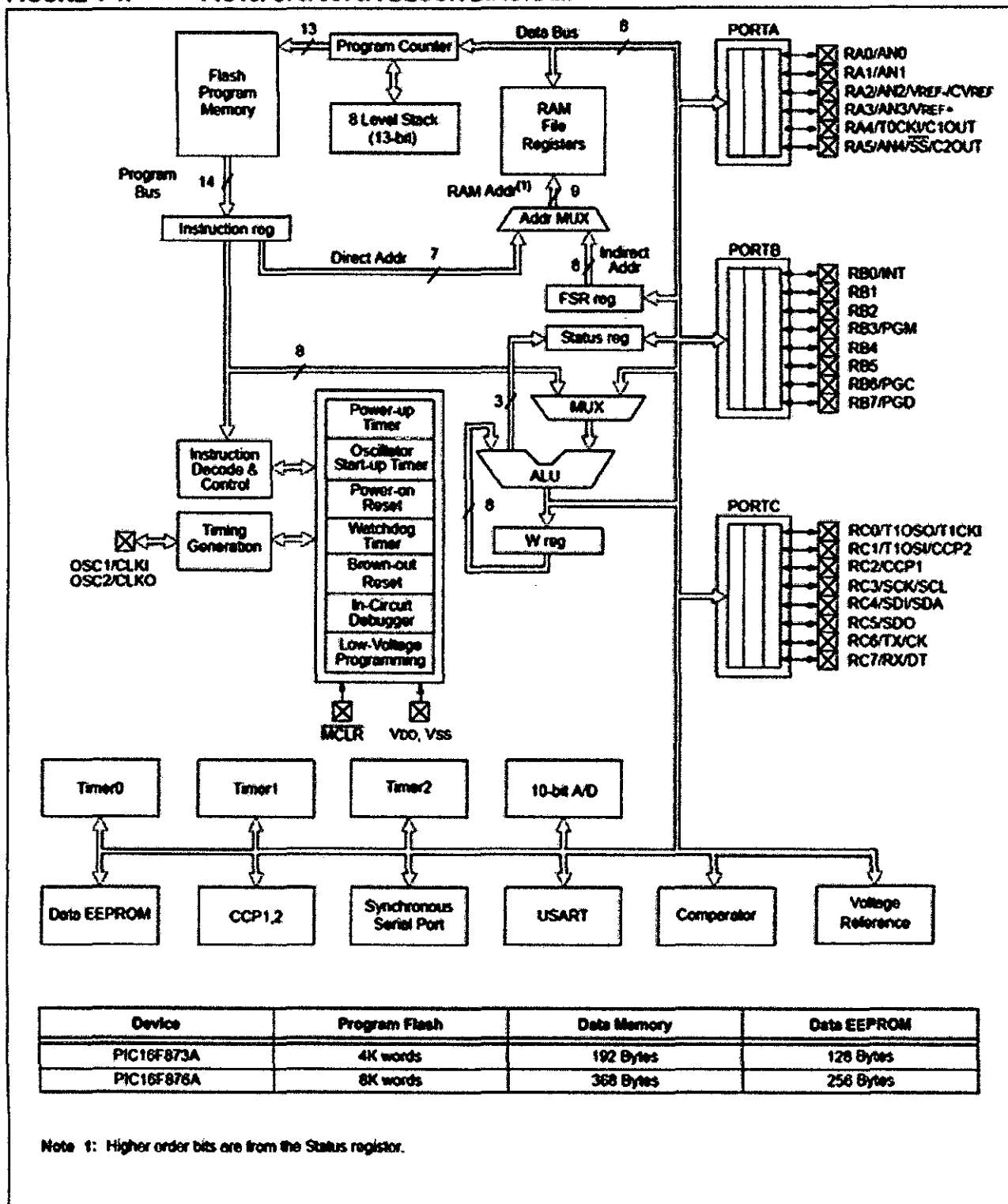


FIGURE 1-2: PIC16F874A/877A BLOCK DIAGRAM

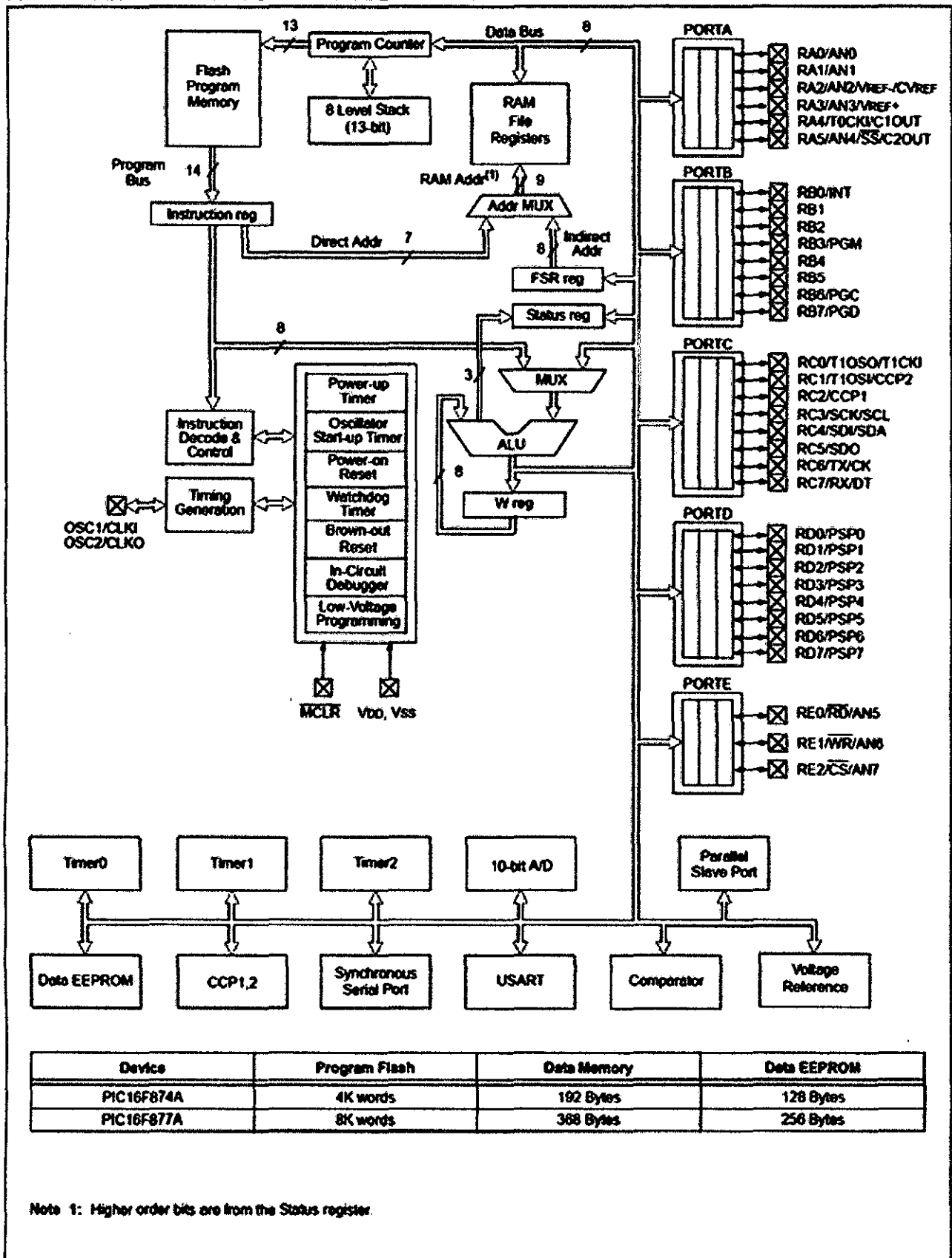


TABLE 1-2: PIC16F873A/876A PINOUT DESCRIPTION

Pin Name	PDIP, SOIC, SSOP Pins	QFN Pins	I/O/P Type	Buffer Type	Description
OSC1/CLKI OSC1 CLKI	9	6	I I	ST/CMOS ⁽³⁾	Oscillator crystal or external clock input. Oscillator crystal input or external clock source input. ST buffer when configured in RC mode; otherwise CMOS. External clock source input. Always associated with pin function OSC1 (see OSC1/CLKI, OSC2/CLKO pins).
OSC2/CLKO OSC2 CLKO	10	7	O O	—	Oscillator crystal or clock output. Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. In RC mode, OSC2 pin outputs CLKO, which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate.
MCLR/VPP MCLR VPP	1	26	I P	ST	Master Clear (input) or programming voltage (output). Master Clear (Reset) input. This pin is an active low Reset to the device. Programming voltage input.
RA0/AN0 RA0 AN0	2	27	I/O I	TTL	PORTA is a bidirectional I/O port. Digital I/O. Analog input 0.
RA1/AN1 RA1 AN1	3	28	I/O I	TTL	
RA2/AN2/VREF-/ CVREF RA2 AN2 VREF- CVREF	4	1	I/O I I O	TTL	
RA3/AN3/VREF+ RA3 AN3 VREF+	5	2	I/O I I	TTL	
RA4/T0CKI/C1OUT RA4 T0CKI C1OUT	6	3	I/O I O	ST	
RA5/AN4/SS/C2OUT RA5 AN4 SS C2OUT	7	4	I/O I I O	TTL	

Legend: I = input O = output I/O = input/output P = power

— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.

2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.

3: This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.

TABLE 1-2: PIC16F873A/876A PINOUT DESCRIPTION (CONTINUED)

Pin Name	PDIP, SOIC, SSOP Pin#	QFN Pin#	I/O/P Type	Buffer Type	Description
RB0/INT RB0 INT	21	18	I/O I	TTL/ST ⁽¹⁾	PORTB is a bidirectional I/O port. PORTB can be software programmed for internal weak pull-ups on all inputs. Digital I/O. External interrupt.
RB1	22	19	I/O	TTL	Digital I/O.
RB2	23	20	I/O	TTL	Digital I/O.
RB3/PGM RB3 PGM	24	21	I/O I	TTL	Digital I/O. Low-voltage (single-supply) ICSP programming enable pin.
RB4	25	22	I/O	TTL	Digital I/O.
RB5	26	23	I/O	TTL	Digital I/O.
RB6/PGC RB6 PGC	27	24	I/O I	TTL/ST ⁽²⁾	Digital I/O. In-circuit debugger and ICSP programming clock.
RB7/PGD RB7 PGD	28	25	I/O I/O	TTL/ST ⁽²⁾	Digital I/O. In-circuit debugger and ICSP programming data.
RC0/T1OSO/T1CKI RC0 T1OSO T1CKI	11	8	I/O O I	ST	PORTC is a bidirectional I/O port. Digital I/O. Timer1 oscillator output. Timer1 external clock input.
RC1/T1OSI/CCP2 RC1 T1OSI CCP2	12	9	I/O I I/O	ST	Digital I/O. Timer1 oscillator input. Capture2 input. Compare2 output. PWM2 output.
RC2/CCP1 RC2 CCP1	13	10	I/O I/O	ST	Digital I/O. Capture1 input. Compare1 output. PWM1 output.
RC3/SCK/SCL RC3 SCK SCL	14	11	I/O I/O I/O	ST	Digital I/O. Synchronous serial clock input/output for SPI mode. Synchronous serial clock input/output for I ² C mode.
RC4/SDI/SDA RC4 SDI SDA	15	12	I/O I I/O	ST	Digital I/O. SPI data in. I ² C data I/O.
RC5/SDO RC5 SDO	16	13	I/O O	ST	Digital I/O. SPI data out.
RC6/TX/CK RC6 TX CK	17	14	I/O O I/O	ST	Digital I/O. USART asynchronous transmit. USART1 synchronous clock.
RC7/RX/DT RC7 RX DT	18	15	I/O I I/O	ST	Digital I/O. USART asynchronous receive. USART synchronous data.
VSS	8, 19	5, 6	P	—	Ground reference for logic and I/O pins.
VDD	20	17	P	—	Positive supply for logic and I/O pins.

Legend: I = input O = output I/O = input/output P = power
 — = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
 2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
 3: This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.

TABLE 1-3: PIC16F874A/877A PINOUT DESCRIPTION

Pin Name	PDIP Pins	PLCC Pins	TQFP Pins	QFN Pins	I/O/P Type	Buffer Type	Description
OSC1/CLKI OSC1 CLKI	13	14	30	32	I I	ST/CMOS ⁽¹⁾	Oscillator crystal or external clock input. Oscillator crystal input or external clock source input. ST buffer when configured in RC mode; otherwise CMOS. External clock source input. Always associated with pin function OSC1 (see OSC1/CLKI, OSC2/CLKO pins).
OSC2/CLKO OSC2 CLKO	14	15	31	33	O O	—	Oscillator crystal or clock output. Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. In RC mode, OSC2 pin outputs CLKO, which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate.
MCLR/VPP MCLR VPP	1	2	18	18	I P	ST	Master Clear (input) or programming voltage (output). Master Clear (Reset) input. This pin is an active low Reset to the device. Programming voltage input.
RA0/AN0 RA0 AN0 RA1/AN1 RA1 AN1 RA2/AN2/VREF-/CVREF RA2 AN2 VREF- CVREF RA3/AN3/VREF+ RA3 AN3 VREF+ RA4/T0CKI/C1OUT RA4 T0CKI C1OUT RA5/AN4/SS/C2OUT RA5 AN4 SS C2OUT	2 3 4 5 6 7	3 4 5 6 8	19 20 21 22 23 24	19 20 21 22 23 24	I/O I I/O I I/O I I O I O I/O I I O	TTL TTL TTL TTL ST TTL	PORTA is a bidirectional I/O port. Digital I/O. Analog input 0. Digital I/O. Analog input 1. Digital I/O. Analog input 2. A/D reference voltage (Low) input. Comparator VREF output. Digital I/O. Analog input 3. A/D reference voltage (High) input. Digital I/O – Open-drain when configured as output. Timer0 external clock input. Comparator 1 output. Digital I/O. Analog input 4. SPI slave select input. Comparator 2 output.

Legend: I = input O = output I/O = input/output P = power
 — = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.

2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.

3: This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.

TABLE 1-3: PIC16F874A/877A PINOUT DESCRIPTION (CONTINUED)

Pin Name	PDIP Pin#	PLCC Pin#	TQFP Pin#	QFN Pin#	I/O/P Type	Buffer Type	Description
RB0/INT RB0 INT	33	36	8	9	I/O I	TTL/ST ⁽¹⁾	PORTB is a bidirectional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs. Digital I/O. External interrupt.
RB1	34	37	9	10	I/O	TTL	Digital I/O.
RB2	35	38	10	11	I/O	TTL	Digital I/O.
RB3/PGM RB3 PGM	36	39	11	12	I/O I	TTL	Digital I/O. Low-voltage ICSP programming enable pin.
RB4	37	41	14	14	I/O	TTL	Digital I/O.
RB5	38	42	15	15	I/O	TTL	Digital I/O.
RB6/PGC RB6 PGC	39	43	16	16	I/O I	TTL/ST ⁽²⁾	Digital I/O. In-circuit debugger and ICSP programming clock.
RB7/PGD RB7 PGD	40	44	17	17	I/O I/O	TTL/ST ⁽²⁾	Digital I/O. In-circuit debugger and ICSP programming data.

Legend: I = input O = output I/O = input/output P = power
 — = Not used TTL = TTL input ST = Schmitt Trigger input

- Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
 2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
 3: This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.

MQ-2 Semiconductor Sensor for Combustible Gas

Sensitive material of MQ-2 gas sensor is SnO_2 , which with lower conductivity in clean air. When the target combustible gas exist, The sensor's conductivity is more higher along with the gas concentration rising. Please use simple electrocircuit, Convert change of conductivity to correspond output signal of gas concentration.

MQ-2 gas sensor has high sensitivity to LPG, Propane and Hydrogen, also could be used to Methane and other combustible steam, it is with low cost and suitable for different application.

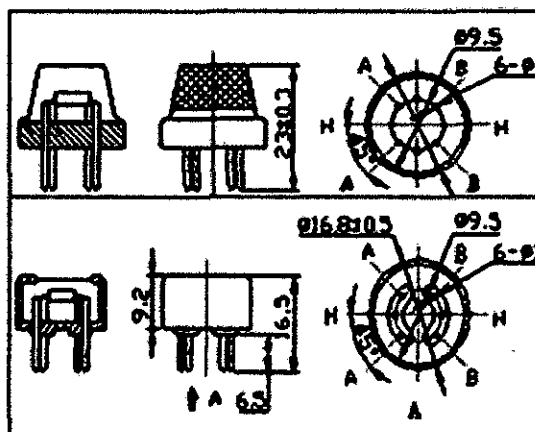
Character

- * Good sensitivity to Combustible gas in wide range
- * High sensitivity to LPG, Propane and Hydrogen
- * Long life and low cost
- * Simple drive circuit

Application

- * Domestic gas leakage detector
- * Industrial Combustible gas detector
- * Portable gas detector

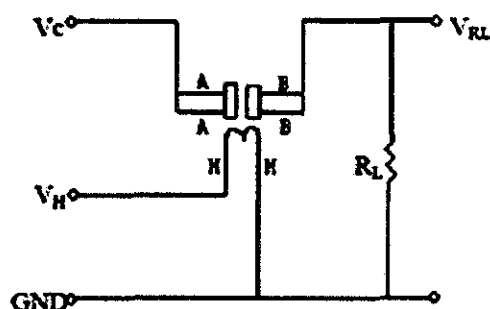
Configuration



Technical Data

Model No.		MQ-2	
Sensor Type		Semiconductor	
Standard Encapsulation		Bakelite (Black Bakelite)	
Detection Gas		Combustible gas and smoke	
Concentration		300-10000ppm (Combustible gas)	
Circuit	Loop Voltage	V_c	$\leq 24V$ DC
	Heater Voltage	V_H	$5.0V \pm 0.2V$ AC or DC
	Load Resistance	R_L	Adjustable
Character	Heater Resistance	R_H	$31\Omega \pm 3\Omega$ (Room Tem.)
	Heater consumption	P_H	$\leq 900mW$
	Sensing Resistance	R_s	$2K\Omega - 20K\Omega$ (in 2000ppm C_2H_6)
	Sensitivity	S	$R_s(\text{in air})/R_s(1000ppm \text{ isobutane}) \geq 5$
	Slope	α	$\leq 0.6(R_{5000ppm}/R_{3000ppm} CH_4)$
Condition	Tem. Humidity	$20^\circ C \pm 2^\circ C$; $65\% \pm 5\% RH$	
	Standard test circuit	$V_c: 5.0V \pm 0.1V$ $V_H: 5.0V \pm 0.1V$	
	Preheat time	Over 48 hours	

Basic test loop



The above is basic test circuit of the sensor. The sensor need to be put 2 voltage, heater voltage(V_H) and test voltage(V_C). V_H used to supply certified working temperature to the sensor, while V_C used to detect voltage (V_{RL}) on load resistance (R_L) whom is in series with sensor. The sensor has light polarity, V_c need DC power. V_C and V_H could use same power circuit with precondition to assure performance of sensor. In order to make the sensor with better performance, suitable R_L value is needed:
Power of Sensitivity body(P_s):
$$P_s = V_c^2 \times R_s / (R_s + R_L)^2$$

Resistance of sensor(R_s): $R_s = (V_c/V_{RL} - 1) \times R_L$

Sensitivity Characteristics

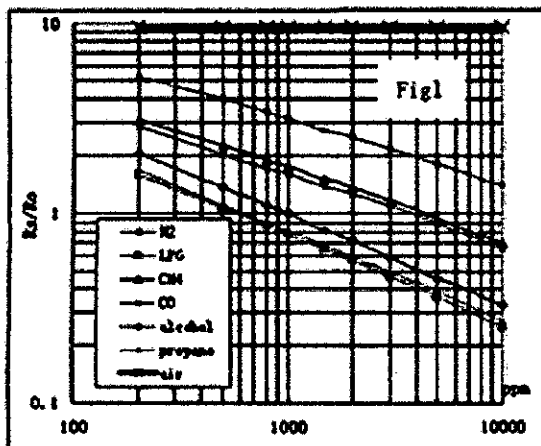


Fig.1 shows the typical sensitivity characteristics of the MQ-2, ordinate means resistance ratio of the sensor (R_s/R_o), abscissa is concentration of gases. R_s means resistance in different gases, R_o means resistance of sensor in 1000ppm Hydrogen. All test are under standard test conditions.

Influence of Temperature/Humidity

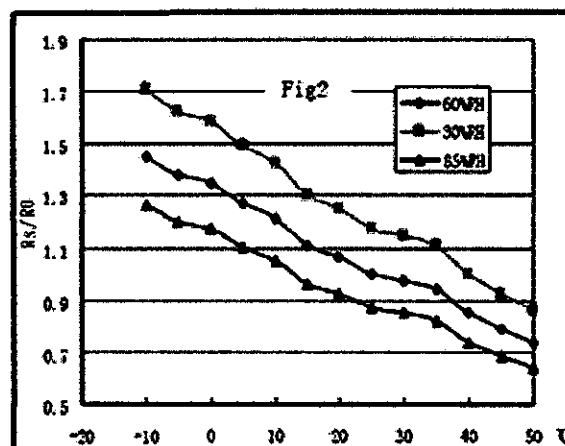
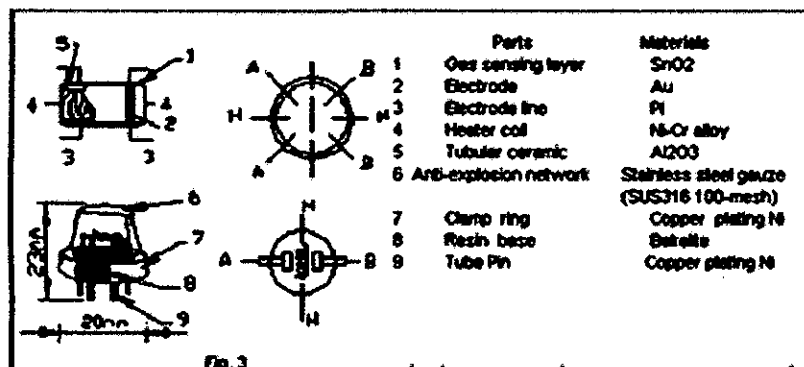


Fig.2 shows the typical temperature and humidity characteristics. Ordinate means resistance ratio of the sensor (R_s/R_o), R_s means resistance of sensor in 1000ppm Butane under different tem. and humidity. R_o means resistance of the sensor in environment of 1000ppm Methane, 20 $^{\circ}C$ /65%RH.

Structure and configuration



Structure and configuration of MQ-2 gas sensor is shown as Fig. 3, sensor composed by micro Al₂O₃ ceramic tube, Tin Dioxide (SnO₂) sensitive layer, measuring electrode and heater are fixed into a crust made by plastic and stainless steel net. The heater provides necessary work conditions for work of sensitive components. The enveloped MQ-2 have 6 pin, 4 of them are used to fetch signals, and other 2 are used for providing heating current.

Notification

1 Following conditions must be prohibited

1.1 Exposed to organic silicon steam

Organic silicon steam cause sensors invalid, sensors must be avoid exposing to silicon bond, fixture, silicon latex, putty or plastic contain silicon environment

1.2 High Corrosive gas

If the sensors exposed to high concentration corrosive gas (such as H_2S , SO_x , Cl_2 , HCl etc), it will not only result in corrosion of sensors structure, also it cause sincere sensitivity attenuation.

1.3 Alkali, Alkali metals salt, halogen pollution

The sensors performance will be changed badly if sensors be sprayed polluted by alkali metals salt especially brine, or be exposed to halogen such as fluorin.

1.4 Touch water

Sensitivity of the sensors will be reduced when splattered or dipped in water.

1.5 Freezing

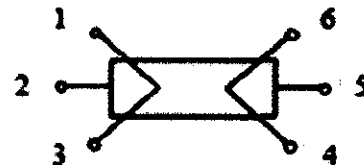
Do avoid icing on sensor's surface, otherwise sensor would lose sensitivity.

1.6 Applied voltage higher

Applied voltage on sensor should not be higher than stipulated value, otherwise it cause down-line or heater damaged, and bring on sensors' sensitivity characteristic changed badly.

1.7 Voltage on wrong pins

For 6 pins sensor, if apply voltage on 1, 3 pins or 4, 6 pins, it will make lead broken, and without signal when apply on 2, 4 pins



2 Following conditions must be avoided

2.1 Water Condensation

Indoor conditions, slight water condensation will effect sensors performance lightly. However, if water condensation on sensors surface and keep a certain period, sensor's sensitivity will be decreased.

2.2 Used in high gas concentration

No matter the sensor is electrified or not, if long time placed in high gas concentration, it will affect sensors characteristic.

2.3 Long time storage

The sensors resistance produce reversible drift if it's stored for long time without electrify, this drift is related with storage conditions. Sensors should be stored in airproof without silicon gel bag with clean air. For the sensors with long time storage but no electrify, they need long aging time for stbility before using.

2.4 Long time exposed to adverse environment

No matter the sensors electrified or not, if exposed to adverse environment for long time, such as high humidity, high temperature, or high pollution etc, it will effect the sensors performance badly.

2.5 Vibration

Continual vibration will result in sensors down-lead response then repture. In transportation or assembling line, pneumatic screwdriver/ultrasonic welding machine can lead this vibration.

2.6 Concussion

If sensors meet strong concussion, it may lead its lead wire disconnected.

2.7 Usage

For sensor, handmade welding is optimal way. If use wave crest welding should meet the following conditions:

2.7.1 Soldering flux: Rosin soldering flux contains least chlorine

2.7.2 Speed: 1-2 Meter/ Minute

2.7.3 Warm-up temperature: $100 \pm 20^\circ C$

2.7.4 Welding temperature: $250 \pm 10^\circ C$

2.7.5 1 time pass wave crest welding machine

If disobey the above using terms, sensors sensitivity will be reduced.

Bluetooth Modules

Wireless UART Cable Replacement

Hardware Manual &
AT Commands Reference Manual Rev. 1r0

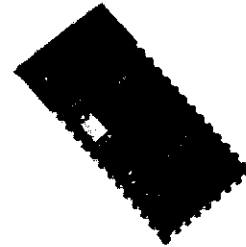
EGBT-045MS and EGBT-046S Bluetooth Module are low cost replacements of our now retired EGBC-04 Bluetooth Module. EGBC-04 is an excellent Bluetooth Module, it is fully certified to Bluetooth standards, and is loaded with programmable features users had come to love. There is just one thing that went against it- it is expensive.

It is easy to see why the EGBC-04 cost so much. Firstly, the manufacturer produced these specialty modules in relatively small volume; hence, there is no economy of the scale to speak of. Secondly, certification costs a lot of money; and this cost will have to be added on top of the manufacturing cost. Hence, EGBC-04 ended up costing about 10 times more expensive than its garden variety USB-type Bluetooth dongles cousins.

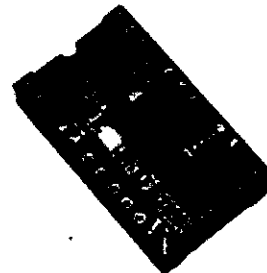
Fortunately, at least one volume manufacturer have came up with an idea of producing a generic Bluetooth module in large quantity, for sale and distribution to developers who now have to put only the firmware functionalities. This resulted in a huge drop in prices of these specialty Bluetooth modules, benefiting us experimenters and hobbyists.

EGBT-045MS and EGBT-046S are generic Bluetooth Modules loaded with SPP firmware for UART wireless cable replacement functions. The EGBT-045MS can be configured by the user to work either as a master or slave Bluetooth device using a set of AT commands.

EGBT-046S, on the other hand, is permanently programmed as Bluetooth slave device. EGBT-046S, because of its simpler function, is a lot easier to use, and of course, costs less than EGBT-045MS. You can use it straight out of the box as a UART wireless cable replacement, without any need to add set-up codes in your microcontroller application



The new EGBT-04 Bluetooth module comes in two flavors. The EGBT-046S is permanently configured as a slave device. EGBT-045MS, on the other hand, can be configured by the user to work as a master or slave Bluetooth device.



EGBT-04 modules can be soldered directly on a hi-rel type IC socket to make it easier to work with prototyping platforms, such as breadboards and perforated prototyping boards.

firmware.

Use the cheaper EGBT-046S if your application will connect to a master Bluetooth device, such as PC or laptops. Use the EGBT-045MS if your application must connect to a slave Bluetooth device, such as with EGBT-046S. Note that EGBT-045MS will work as well as a slave Bluetooth device.

COMMON SPECIFICATIONS

Radio Chip: CSR BC417
 Memory: External 8Mbit Flash
 Output Power: -4 to +6dbm Class 2
 Sensitivity: -80dbm Typical
 Bit Rate: EDR, up to 3Mbps
 Interface: UART
 Antenna: Built-in
 Dimension: 27W x 13H mm

Voltage: 3.1 to 4.2VDC
 Current: 40mA max

COMMON HARDWARE INTERFACING CONSIDERATIONS

The EGBT-04 module will work with supply voltage of 3.1VDC to 4.2VDC. When supplied with 3.3VDC, it will interface directly with the UART port of any microcontroller chip running at 3.3VDC.

When used with 5V microcontrollers, The TXD output logic swing of the EGBT-04 still falls within the valid 5V TTL range, hence, can be connected directly to the UART RXD of the 5V microcontroller host. EGBT RXD and inputs, however, are not 5V tolerant, and can be damaged by 5V level logic going in. Some level translation circuit must be added to protect the inputs.

A simple diode level translator circuit like the ones shown in Figure 3 and 7 will suffice in most applications. A better alternative is with the use of 5V input tolerant tiny logic chips such as 74LVC1G125 – a single buffer chip housed in smd sot23-5 package.

EGBT-046S PIN CONFIGURATION

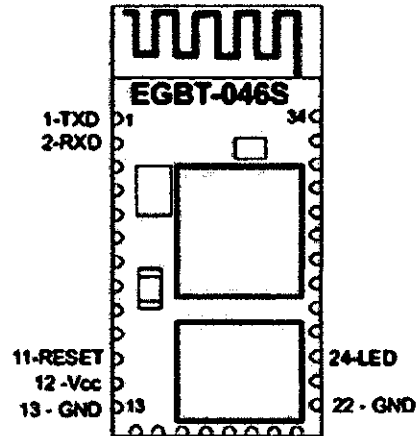


Figure 1. EGBT-046S Pin Layout

Table 1. EGBT-046S Pin Description

PIN	ID	DESCRIPTION
1	TXD	UART TXD Output
2	RXD	UART RXD Input
11	RESET	RESET Input
12	Vcc	+3.1 to 4.2VDC Power Input
13	GND	Common Ground
22	GND	Common Ground
24	LED	LED Status Indicator Flashing - Waiting to Connect/Pair Steady ON - Connected/Paired

Note:

All unassigned pins must be left unconnected.

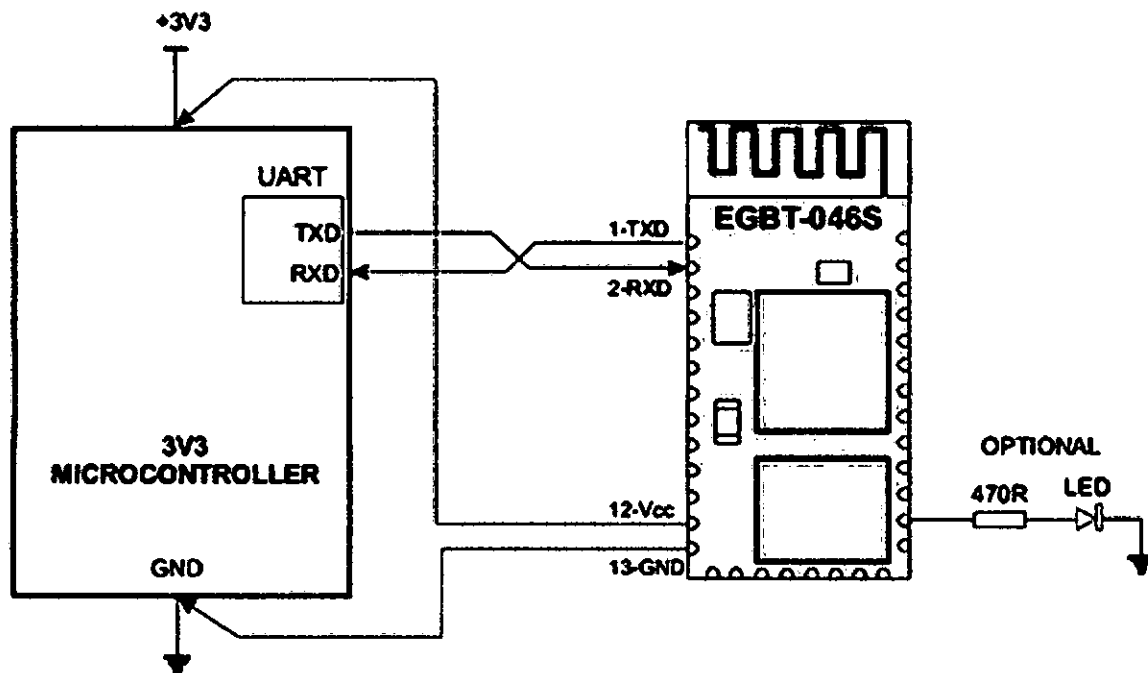


Figure 2. EGBT-046S wiring example with a 3v3 host microcontroller. The 470R resistor and LED are for status indication, and may be omitted if not needed.

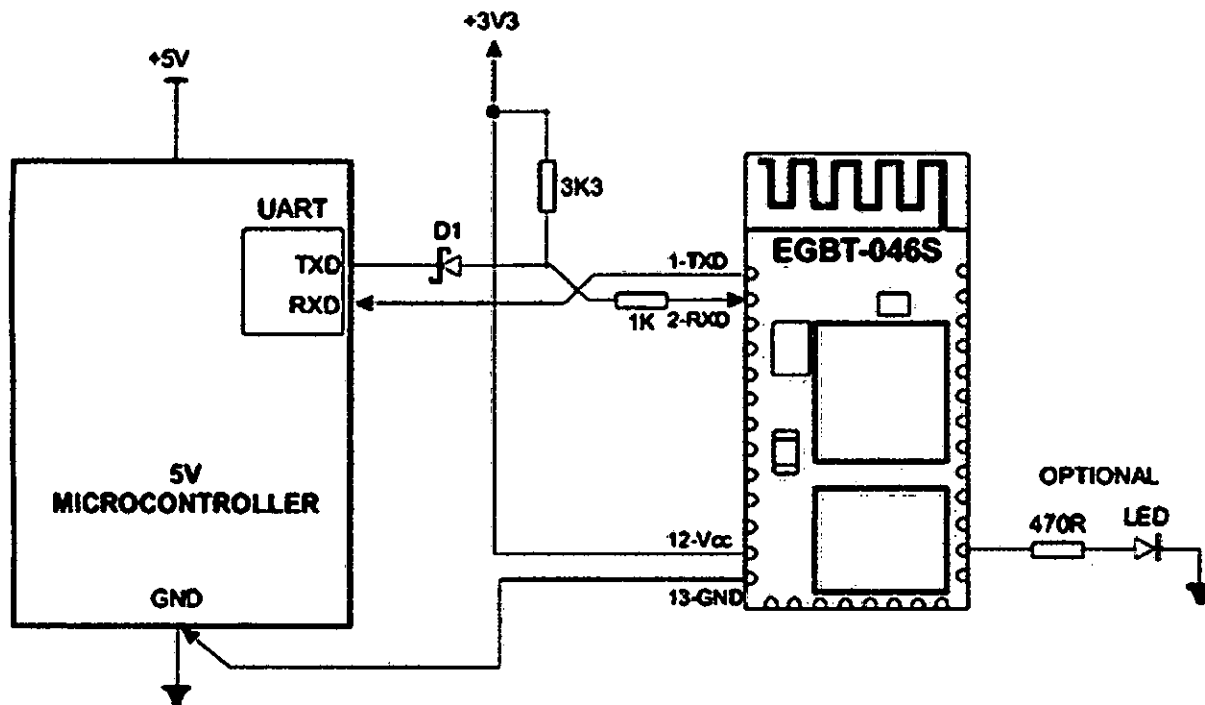


Figure 3. EGBT-046S RX input is not 5V tolerant. A schottky diode connected as shown will keep 5V voltages out of the Bluetooth module when operated with a 5V host microcontroller.

PREPARATION FOR USE

The EGBT-046S is permanently configured as a slave Bluetooth device. It works under the following default configuration:

Baud Rate: 9600 bps
Data : 8 bits
Stop Bits: 1 bit
Parity : None
Handshake: None

Passkey: 1234
Device Name: linvor

If the default configuration suits your application, then you can use EGBT-046S immediately. Once it is paired to a master Bluetooth device, its operation becomes transparent to the user. No user code specific to the Bluetooth module is needed at all in the user microcontroller program.

The EGBT-046S automatically sets itself up in Command Mode when it is not remotely connected.

(paired) to any other Bluetooth device. You can change the Passkey, Device Name, and Baud Rate while the EGBT-046S is in Command Mode by entering a small subset of AT style commands. Any changes made will be retained even after power is removed from the EGBT-046S, hence device configuration setup must not be repeated unless new changes need to be made.

You can do configuration setup using the host controller itself (the microcontroller in your own circuit), or a PC running a terminal software using a serial to TTL (or USB to Serial TTL) converter. See Figure 4 for connection details.

It is important to note that EGBT-046S does not wait for any termination character for each AT command entry. Instead, it acts to whatever character you entered after one second. Hence, if you are not able to complete a command entry within a second, it will be ignored. Because of this behavior, it may be extremely difficult to do manual entry configuration using Windows Hyperterminal software. Terminal software that allows batch sending of multiple characters must be used.

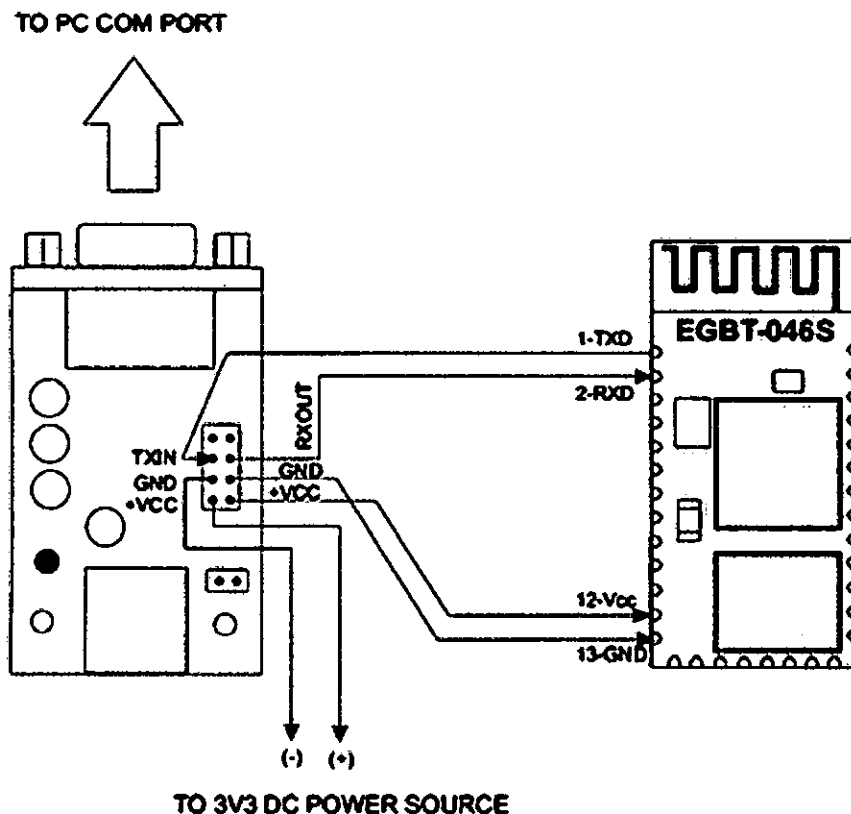


Figure 4. A PC may be used to configure the EGBT-04. To connect to a PC COM port, a RS-232C to TTL converter is needed. This figure shows a wiring example using e-Gizmo RS-232 to TTL converter kit.

EGBT-046S AT Command Set

1. TEST

Used to test the UART connection between the host controller and Bluetooth Module.

COMMAND	RESPONSE
AT	OK

2. Change Baud Rate

COMMAND	RESPONSE
AT+BAUD<p>	OK<r>

where:

<p> Paramater

<r> Response, set to nnnn bps

<p>	<r>	Remarks
1	1200	set to 1200bps
2	2400	set to 2400bps
3	4800	set to 4800bps
4	9600	set to 9600bps (Default)
5	19200	set to 19200bps
6	38400	set to 38400bps
7	57600	set to 57600bps
8	115200	set to 115200bps
9	230400	set to 230400bps
A	460800	set to 460800bps
B	921600	set to 921600bps
C	1382400	set to 1382400bps

Caution:

PC standard COM port hardware does not support baud rates in excess of 115200bps. If you are using a PC to configure EGBT-046S and accidentally set baud rate to these values, connection to a PC COM port will no longer be possible. Use of USB to Serial converter cable that can work at higher bauds may be necessary to re-establish a connection. Prolific PL-2303 based USB to Serial converter cables are known to work up to 921600bps.

Example1: Set baud rate to 57600bps

From Host controller:

AT+BAUD7

EGBT-046S Response

OK57600

Example2: Set baud rate to 4800bps

From Host controller:

AT+BAUD3

EGBT-046S Response

OK4800

3. Change Device Name

The EGBT-046S can be assigned a readable name of up to 20 characters in lenght.

COMMAND	RESPONSE
AT+NAME<name>	OK<name>

Example1: Set device name as EGBT-04

From Host controller:

AT+NAMEEGBT-04

EGBT-046S Response

OKEGBT-04

4. Change PASSKEY(PIN code)

Passkey (PIN Code) is a 4-digit code shared with a master Bluetooth Device (e.g. PC) to prevent unauthorized pairing.

COMMAND	RESPONSE
AT+PIN<nnnn>	OK<nnnn>

Where:

<nnnn> 4-digit passkey

Example1: Set PASSKEY to 5995

From Host controller:

AT+PIN5995

EGBT-046S Response

OK5995

DATASHEET DEL MÓDULO SIM900

(1) The SIM900 does work, but deviations from the GSM specification may occur.

Table 4: Coding schemes and maximum net data rates over air interface

CS-1:	9.05kbps	18.1kbps	36.2kbps
CS-2:	13.4kbps	26.8kbps	53.6kbps
CS-3:	15.6kbps	31.2kbps	62.4kbps
CS-4:	21.4kbps	42.8kbps	85.6kbps

2.2 SIM900 Functional Diagram

The following figure shows a functional diagram of the SIM900 and illustrates the mainly functional part:

- The GSM baseband engine
- Flash and SRAM
- The GSM radio frequency part
- The antenna interface
- The Other interfaces

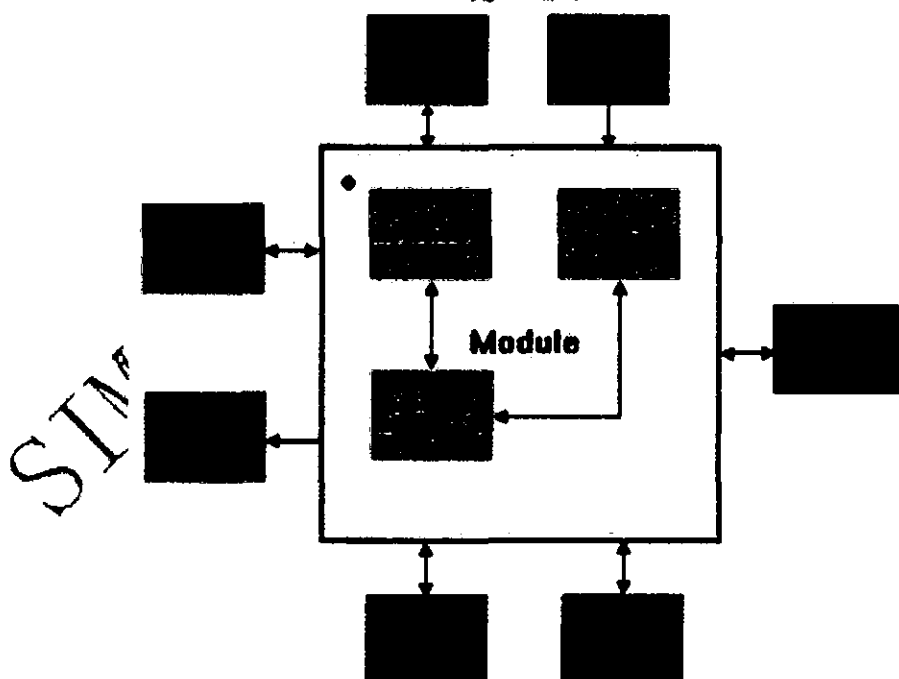


Figure 1: SIM900 functional diagram

2.3 SIM900 Evaluation Board

In order to help you on the application of SIM900, SIMCom can supply an Evaluation Board (EVB) that interfaces the SIM900 directly with appropriate power supply, SIM card holder, RS232 serial port, handset port, earphone port, line in port, antenna and all GPIO of the SIM900.

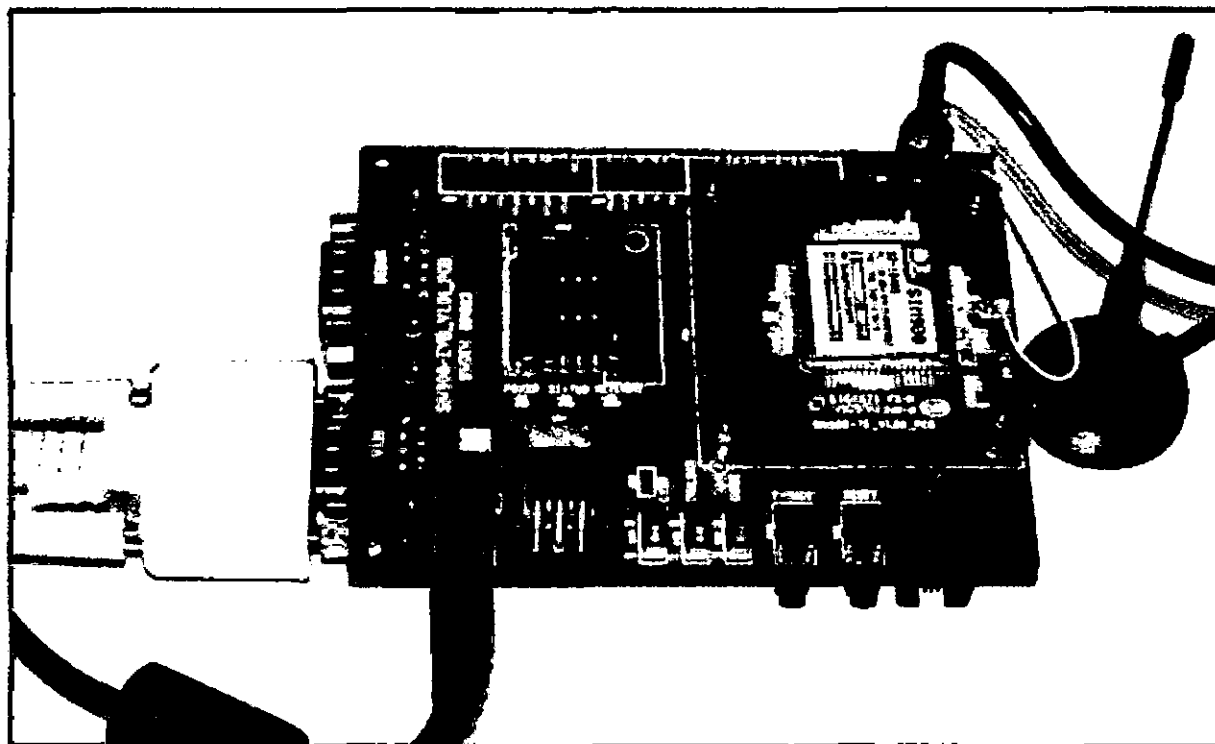


Figure 2: Top view of SIM900 EVB

For details please refer to the *SIM900-EVB_UGD* document.

3 Application Interface

SIM900 is equipped with a 68-pin SMT pad that connects to the cellular application platform. Sub-interfaces included in these SMT pads are described in detail in following chapters:

- Power supply (*please refer to Chapter 3.3*)
- Serial interfaces (*please refer to Chapter 3.8*)
- Analog audio interfaces (*please refer to Chapter 3.9*)
- SIM interface (*please refer to Chapter 3.10*)

Electrical and mechanical characteristics of the SMT pad are specified in Chapter 5.

3.1 SIM900 Pin Description

Table 5: Pin description

PIN NAME	I/O	DESCRIPTION	DC CHARACTERISTICS	COMMENT
VBAT	I	3 VBAT pins are dedicated to connect the supply voltage. The power supply of SIM900 has to be a single voltage source of VBAT= 3.1V...4.8V. It must be able to provide sufficient current in a transmit burst which typically rises to 2A.	Vmax= 4.8V Vmin=3.1V Vnorm=4.0V	
VRTC	I/O	Current input for RTC when the battery is not supplied for the system. Current output for backup battery when the main battery is present and the backup battery is in low voltage state.	Vmax=3.15V Vmin=2.0V Vnorm=3.0V Iout(max)= 300uA Iin=2 uA	If RTC backup is needed, a battery or capacitor should be connected with the VRTC pin. If RTC backup is not needed, a capacitor should be connected to this pin.

VDD_EXT	O	2.8V output power supply	V _{max} =2.95V V _{min} =2.70V V _{norm} =2.80V I _{out(max)} = 10mA	If unused, keep open.
GND		Ground		

Power on or power off

PIN NAME	I/O	DESCRIPTION	DC CHARACTERISTICS	
PWRKEY	I	Voltage input for PWRKEY. PWRKEY should be pulled low to power on or power off the system. The user should drive the PWRKEY to low level voltage for a short time when power on or power off the system because the system need margin time in order to assert the software.	V _{ILmax} =0.15*VDD_EXT V _{IHmin} =0.85* VDD_EXT V _{Imax} =VDD_EXT V _{ILmin} = 0V	It is already pulled up.
PWRKEY_OUT	O	Connecting PWRKEY and PWRKEY_OUT for a short time then release also can power on or power off the module.	V _{OHmin} = VDD_EXT-0.1V V _{OLmax} =0.1V V _{OHmax} = VDD_EXT V _{OLmin} = 0V	

Audio interfaces

PIN NAME	I/O	DESCRIPTION	DC CHARACTERISTICS	COMMENT
MIC_P MIC_N	I	Positive and negative voice band input	Audio DC Characteristics refer to chapter 3.9	If unused keep open
SPK_P SPK_N	O	Positive and negative voice band output		If unused keep open
LINEIN_R LINEIN_L	I	Line input		If unused keep open

GENERAL PURPOSE input/output

PIN NAME	I/O	DESCRIPTION	DC CHARACTERISTICS	COMMENT
STATUS	O	Indicate working status	V _{ILmax} =0.15 *VDD_EXT V _{IHmin} =0.85*VDD_EXT	If unused keep open
NETLIGHT	O	Indicate net status	V _{ILmin} = 0V V _{IHmax} = VDD_EXT	If unused keep open
DISP_DATA	I/O	Display interface	V _{OHmin} = VDD_EXT-0.1V	If unused keep open
DISP_CLK	O		V _{OLmax} =0.1V	
DISP_CS	O		V _{OHmax} = VDD_EXT	
DISP_D/C	O		V _{OLmin} = 0V	

SCL	O	I ² C bus clock		If unused keep open
SDA	I/O	I ² C bus data		
KBR0-KBR4	O	Keypad interface		If unused keep these pins open
KBC0-KBC4	I			Pull up to VDD_EXT. if unused keep pins open

PIN NAME	I/O	DESCRIPTION	DC CHARACTERISTICS	COMMENT
RXD	I	Receive data	$V_{ILmax}=0.15 * VDD_EXT$ $V_{IHmin}=0.85 * VDD_EXT$ $V_{ILmin}=0V$ $V_{IHmax}= VDD_EXT$ $VOHmin= VDD_EXT-0.1V$ $VOLmax=0.1V$ $VOHmax= VDD_EXT$ $VOLmin= 0V$	RXD pin should be pulled up to VDD_EXT. DTR Pin has been pulled up to VDD_EXT internally. Other unused pins can keep open
TXD	O	Transmit data		
RTS	I	Request to send		
CTS	O	Clear to send		
RI	O	Ring indicator		
DSR	O	Data Set Ready		
DCD	O	Data carry detect		
DTR	I	Data terminal Ready		

PIN NAME	I/O	DESCRIPTION	DC CHARACTERISTICS	COMMENT
DBG_TXD	O	Serial interface for debugging and firmware upgrade	$V_{ILmax}=0.15 * VDD_EXT$ $V_{IHmin}=0.85 * VDD_EXT$ $V_{ILmin}=0V$ $V_{IHmax}= VDD_EXT$ $VOHmin= VDD_EXT-0.1V$ $VOLmax=0.1V$ $VOHmax= VDD_EXT$ $VOLmin= 0V$	If unused keep pins open
DBG_RXD	I			

PIN NAME	I/O	DESCRIPTION	DC CHARACTERISTICS	COMMENT
SIM_VDD	O	Voltage supply for SIM card	The voltage can be select by software automatically either 1.8V or 3V	
SIM_DATA	I/O	SIM data output	$V_{ILmax}=0.15 * SIM_VDD$ $V_{IHmin}=0.85 * SIM_VDD$ $V_{ILmin}= 0V$ $V_{IHmax}= SIM_VDD$	All signals of SIM interface are protected against ESD
SIM_CLK	O	SIM clock		
SIM_RST	O	SIM reset		

			$VOH_{min} = SIM_VDD - 0.1V$ $VOL_{max} = 0.1V$ $VOH_{max} = SIM_VDD$ $VOL_{min} = 0V$	with a TVS diode array. Maximum cable length 200mm from the module pad to SIM card holder.
SIM_PRESENCE	I	SIM detect	$VIL_{max} = 0.15 * VDD_EXT$ $VIH_{min} = 0.85 * VDD_EXT$ $VIL_{min} = 0V$ $VIH_{max} = VDD_EXT$	If unused keep open

PIN NAME	I/O	DESCRIPTION	DC CHARACTERISTICS	COMMENT
ADC	I	General purpose analog to digital converter.	Input voltage range: 0V ~ 2.8V	If unused keep open

PIN NAME	I/O	DESCRIPTION	DC CHARACTERISTICS	COMMENT
NRESET	I	External reset input(Active low)	$VIL_{max} = 0.15 * VDD_EXT$ $VIH_{min} = 0.85 * VDD_EXT$ $VIL_{min} = 0V$ $VIH_{max} = VDD_EXT$	If unused keep open

PIN NAME	I/O	DESCRIPTION	DC CHARACTERISTICS	COMMENT
PWM1	O	PWM Output	$VOH_{min} = VDD_EXT - 0.1V$ $VOL_{max} = 0.1V$	If unused keep open
PWM2	O	PWM Output	$VOH_{max} = VDD_EXT$ $VOL_{min} = 0$	

3.2 Operating Modes

The table below briefly summarizes the various operating modes referred to in the following chapters.

Table 6: Overview of operating modes

Mode	Feature	Description
Normal operation	GSM/GPRS SLEEP	Module will automatically go into SLEEP mode if DTR is set to high level and there is no on air and no hardware interrupt (such as GPIO interrupt or data on serial port).

		In this case, the current consumption of module will reduce to the minimal level. In SLEEP mode, the module can still receive paging message and SMS from the system normally.
	GSM IDLE	Software is active. Module has registered to the GSM network, and the module is ready to send and receive.
	GSM TALK	Connection between two subscribers is in progress. In this case, the power consumption depends on network settings such as DTX off/on, FR/EFR/HR, hopping sequences, antenna.
	GPRS STANDBY	Module is ready for GPRS data transfer, but no data is currently sent or received. In this case, power consumption depends on network settings and GPRS configuration.
	GPRS DATA	There is GPRS data transfer (PPP or TCP or UDP) in progress. In this case, power consumption is related with network settings (e.g. power control level), uplink / downlink data rates and GPRS configuration (e.g. used multi-slot settings).
POWER DOWN		Normal shutdown by sending the "AT+CPOWD=1" command or using the PWRKEY. The power management ASIC disconnects the power supply from the baseband part of the module, and only the power supply for the RTC is remained. Software is not active. The serial port is not accessible. Operating voltage (connected to VBAT) remains applied.
Minimum functionality mode (without remove power supply)		Use the "AT+CFUN" command can set the module to a minimum functionality mode without remove the power supply. In this case, the RF part of the module will not work or the SIM card will not be accessible, or both RF part and SIM card will be closed, and the serial port is still accessible. The power consumption in this case is very low.

3.3 Power Supply

The power supply of SIM900 is from a single voltage source of VBAT= 3.1V...4.8V. In some case, the ripple in a transmitting burst may cause voltage drops when current consumption rises to typical peaks of 2A. So the power supply must be able to provide sufficient current up to 2A.

For the VBAT input, a local bypass capacitor is recommended. A capacitor (about 100 μ F, low ESR) is recommended. Multi-layer ceramic chip (MLCC) capacitors can provide the best combination of low ESR and small size but may not be cost effective. A lower cost choice may be a 100 μ F tantalum capacitor (low ESR) with a small (0.1 μ F to 1 μ F) ceramic in parallel, which is illustrated as following figure. The capacitors should be placed as close as possible to the SIM900 VBAT pins. The following figure is the recommended circuit.

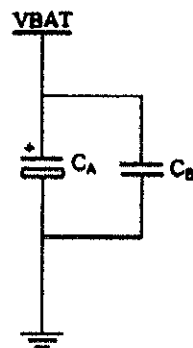


Figure 3: Reference circuit of the VBAT input

The circuit design of the power supply depends strongly upon the power source where this power is drained. The following figure is the reference design of +5V input source power supply. The designed output for the power supply is 4.1V, thus a linear regulator can be used. If there's a big difference between the input source and the desired output (VBAT), a switching converter power supply will be preferable because of its better efficiency especially with the 2A peak current in burst mode of the module.

The single 3.6V Li-Ion cell battery type can be connected to the power supply of the SIM900 VBAT directly. But the Ni_Cd or Ni_MH battery types must be used carefully, since their maximum voltage can rise over the absolute maximum voltage for the module and damage it.

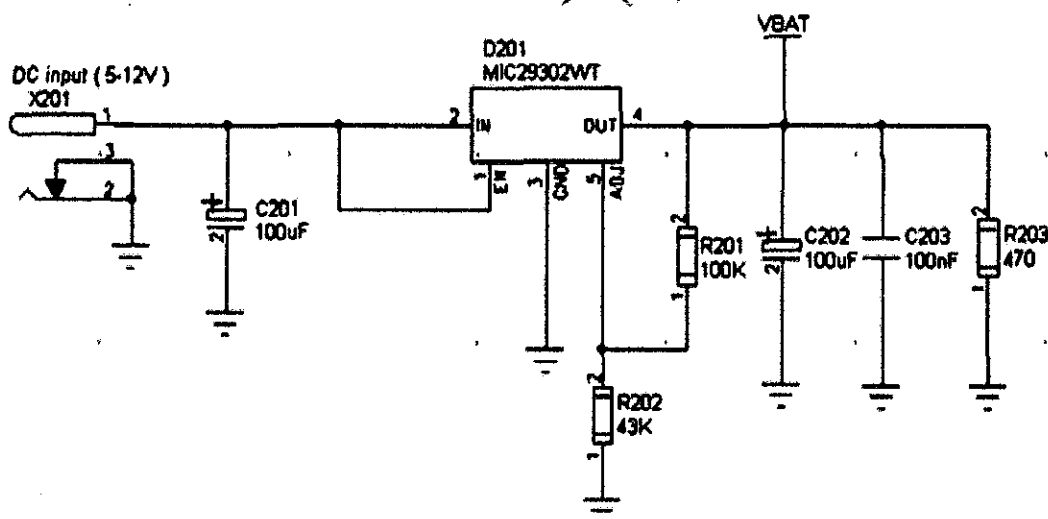


Figure 4: Reference circuit of the source power supply input

The following figure is the VBAT voltage ripple wave at the maximum power transmit phase, the test condition is VBAT=4.0V, VBAT maximum output current =2A, $C_A=100\mu\text{F}$ tantalum capacitor (ESR=0.7 Ω) and $C_B=1\mu\text{F}$.

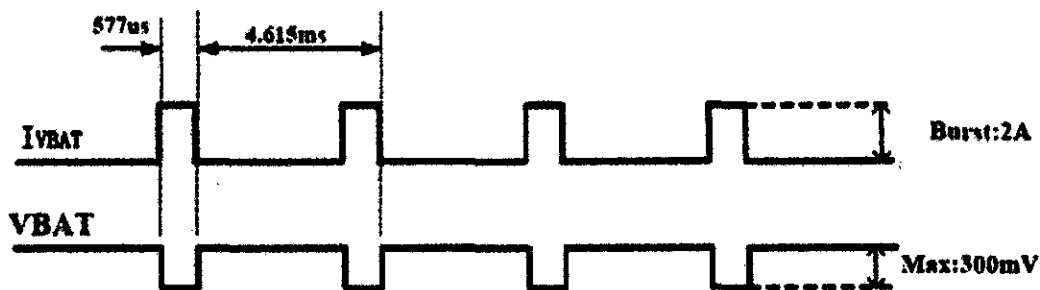


Figure 5: VBAT voltage drop during transmit burst

3.3.1 Power Supply Pins

Three VBAT pins are dedicated to connect the supply voltage and fifteen GND pins are dedicated to connect ground. VRTC pin can be used to back up the RTC.

3.3.2 Minimizing Power Losses

When designing the power supply for your application please pay specific attention to power losses. Ensure that the input voltage VBAT never drops below 3.1V even in a transmit burst where current consumption can rise to typical peaks of 2A. If the power voltage drops below 3.1V, the module may be switched off. The PCB traces from the VBAT pins to the power source must be wide enough to decrease voltage drops in the transmitting burst mode.

3.3.3 Monitoring Power Supply

To monitor the supply voltage, you can use the "AT+CBC" command which include a parameter: voltage value (in mV).

The voltage is continuously measured at intervals depending on the operating mode. The displayed voltage (in mV) is averaged over the last measuring period before the AT+CBC command is executed.

For details please refer to document [1]

3.4 Power Up and Power Down Scenarios

In general, be sure not to turn on SIM900 while it is beyond the safety limits of voltage and temperature stated in Chapter 3.4.2. SIM900 would immediately switch off after having started and detected these inappropriate conditions. In extreme cases this can cause permanent damage to the module.

3.4.1 Turn on SIM900

SIM900 can be turned on by two ways, which are described in following chapters: